D'Fusion® @Home

Deployment Guide

2012-12-14

TABLE OF CONTENTS

TABLE OF FIGURES

1 INTRODUCTION

1.1 Object

This document describes how to use D'Fusion @Home to massively deploy augmented reality applications on PCs and Macs.

This document is a technical reference manual and targets an audience with a technical background and involved directly within the D'Fusion @Home project creation and utilization.

This document is a counterpart for the existing D'Fusion documentation and does not replace them. It provides information specific to massive deployment of D'Fusion @Home projects.

1.2 External documentation

1.2.1 Reference documents

[01] D'Fusion Studio - User Guide, Total Immersion
DFusion Studio - User Guide.pdf

[02] D'Fusion Augmented Reality - Reference Manual, Total Immersion
DFusion AR - Reference Manual.pdf

[03] D'Fusion Augmented Reality - Lua API, Total Immersion
DFusion AR - Lua API.pdf

[04] D'Fusion Augmented Reality - Physics Plugin, Total Immersion
DFusion AR - Physics Plugin.pdf

[05] D'Fusion Computer Vision - Reference Manual, Total Immersion
DFusion CV - Reference Manual.pdf

[06] D'Fusion Exporter for Maya - User Manual, Total Immersion
DFusion Exporter for Maya - User Guide.pdf

[07] D'Fusion Exporter for Maya - Modeling Constraints, Total Immersion
DFusion Exporter for Maya - Modeling Constraints.pdf

[08] D'Fusion Exporter for 3dsMax - User manual, Total Immersion
DFusion Exporter for 3dsMax - User Guide.pdf

[09] D'Fusion Exporter for 3dsMax - Modeling Constraints, Total Immersion
DFusion Exporter for 3dsMax - Modeling Constraints.pdf

1.2.2 Other documents

[10] LUA 5.1 Reference Manual
www.lua.org/manual/5.1

1.3 Glossary & Acronyms

TI	Total Immersion
AR	Augmented Reality
MLT	Marker less Tracking
CV	Computer Vision

Software Key A Software Key is necessary to play a D'Fusion @Home application. A Software Key can be of two types: Test or Final.

Test Software Key A Software Key used to deploy a D'Fusion @Home scenario in **Test mode** (with a watermark). This key is generated when exporting the AR scenario from D' Fusion Studio, or when using the tiProtectorAR2 tool (available to Pro users only).

Final Software Key A Software Key used to deploy a D'Fusion @Home scenario in **Final mode** (without watermark). This key is provided by Total Immersion to D'Fusion Studio Suite Pro users only.

1.4 Presentation rules

Presentation	Content
XXXXXX	File name
XXXXXXX	Script: code samples
XXXXXXX	Key word
XXXXXXX	Reminder: main information of the chapter

2 D'FUSION @HOME PRODUCT DESCRIPTION

D'Fusion @Home is a deployment product available with **D'Fusion® Studio Suite**.

D'Fusion® Studio Suite also includes **D'Fusion Studio**, the software that will allow you to design and develop Augmented Reality scenarios and create the associated media.

D'Fusion @Home will allow you to massively deploy augmented reality applications on a large range of computers (PCs and Macs), and to play the AR scenarios created with D'Fusion Studio.

2.1 @Home Web and @Home Desktop

There are two versions of D'Fusion @Home:

- **D'Fusion @Home Web** – for web-based AR applications: the final user uses a web-browser to play the augmented reality application, connected to a web-page on a distant web-server.
- **D'Fusion @Home Desktop** – for standalone AR applications: no web browser and no internet connection are required to play the augmented reality application.

Each version contains all the libraries, plug-ins and players required for @Home Web and @Home Desktop deployment, respectively.

Depending on the final deployment of your AR application you may be using one or the other. The workflow described below is valid for both versions.

2.2 Deployment workflow

The following steps describe the workflow for creating a D'Fusion @Home application:

2.2.1 Create your AR scenario

Use D'Fusion Studio to create your AR scenario.

If you want to ensure the compatibility of your application across a large range of platforms (PCs and Mac), read the recommendations described in section 3 below.

Documentation on how to use D'Fusion Studio to create AR scenarios is available in [01].

2.2.2 Export your project

Export your D'Fusion Studio project using the "@Home" export target setting.

This action creates a folder (the "exported project folder") designed to contain all the files necessary to play your @Home application (AR scenario, media, etc.). Your exported project, exported media and so on are to be found inside this folder.

Documentation on how to use D'Fusion Studio to export AR scenarios is available in [01].

Note: we recommend using encryption when exporting your project.

2.2.3 Launch and test your application

When your project is exported, a Test Software Key (.dfk file) is automatically generated and saved next to your exported project (i.e. in your exported project folder). This Test Software Key will allow you to play your AR application in Test mode with the suitable player: D'Fusion @Home Desktop Player if you are deploying for desktop platforms, or D'Fusion @Home Web Player if you are deploying for the web.

2.2.4 Final deployment

Once your application is tested and validated, you can proceed to final deployment by requesting a Final Software Key from Total Immersion.

If you are a D'Fusion Studio Suite Pro user, you can make this request by sending us your Test Software Key(s) (.dfk file(s)). Total Immersion will send the Final Software Key(s) (.dfk file(s)) back to you.

The Final Software Key will allow you to play your augmented reality application in Final mode (without watermark).

To play your application in Final mode, replace your Test Key(s) with your Final key(s) in your exported project folder(s). Then follow the steps described in chapter 0.

Please note that deployment for commercial use (without a watermark) is only available with D'Fusion Studio Suite Pro.

2.3 Hardware and Software requirements

2.3.1 System configuration

Platform		PC			Mac (Intel architecture)		
OS		Windows XP SP3 32	Windows Seven 32 & 64	Windows 8 * *32 & 64*	MacOS 10.6 Snow Leopard	MacOS 10.7.5 Lion	MacOS 10.8 Mountain Lion
Browser	Internet Explorer	Minimum, Recommended: IE 9 (32 bits only) Maximum IE10 (32 bits only) (10.0.9200.x)			N/A		
	Chrome	Minimum: Chrome 16.0 Maximum, Recommended : Chrome 23.0.1271			N/A		
	Firefox	Minimum: Firefox 15.0 (32 bits only) Maximum: Firefox 16.0.2 (32 bits only) Recommended: Firefox 16.0.2 (32 bits only)			Minimum: 15.0 (32 bits mode) Maximum: Firefox 16.0.2 (32 bits mode) Recommended: Firefox 16.0.2 (32 bits mode)		
	Safari	N/A			Minimum: Safari 5.1.7 ** Maximum: Safari 6.0.1 ** Recommended: 6.0.1** *(No keyboard input)*		
CPU		Minimum: Intel Dual Core (or core 2 duo) 1.5 GHz or AMD equivalent Recommended: Intel Dual Core (or core 2 duo) 2.4 GHz or AMD equivalent					
Memory		Minimum: 1 GB Recommended: 2 GB					
Graphic card		Minimum: Most graphics cards (Nvidia, ATI, Intel chipset, except Via chipset) Recommended: Powerful graphic card (Nvidia, ATI) with 256 MB memory					
Hard disk		Minimum : 100 MB for the software. More is needed for applications					
Camera		Specifications depends on final usage					

( ) Windows 8 editions: Windows 8 Pro, Windows 8 Enterprise*

Figure 1: System requirements

Remarks:

Notebook PCs can be supported, but the scenario must be light (as opposed to complex).

Windows operating systems with the "N" or "KN" tag (European and South Korean zones only) must be updated to be able to run Windows Media Player.

We recommend having the latest Windows update installed.

Mac based on Power PC processor are not supported!

Ongoing maintenance for the D'Fusion @Home Web plugins on Mac OS X internet browsers is not planned for 2013. In other words, Safari and Firefox on Mac OS X will remain compatible insofar as these browsers don't change their event model interface support implementation, and continue to support the Carbon interface.

2.4 Cross-platform features availability

The following tables describe D'Fusion features' availability through various platforms and products. All D'Fusion supported targets have been given for comparison purposes.

D'Fusion Pro is set as a reference. As a result, if a feature is available for D'Fusion Pro, it will appear in the table even if it is not available or only partially available for another platform or product.

Cross platforms features matrix Reference : D'Fusion Pro	@Home Windows	@Home Mac OS X	Mobile Android	Mobile iPhone	for Adobe Flash Player
3D Engine (Ogre 3D)	+	+	+	+	Away3D Lite
Textures	+	+	jpg, png, dds	jpg, png, dds	+
Textures	+	+	Only size power of 2	Only size power of 2	+
Shaders	+	+	+ (OpenGL-ES1 or ES2)	+ (OpenGL-ES1 or ES2)	-
Meshes	+	+	Public only	Public only	+
Shadows	+	+	-	-	-
Envmaps	+	+	+	+	(shader)
2D Overlay	+	+	-	-	-
2D Text	+	+	-	-	-
Particles	+	+	+	+	-
Script Engine	Lua	Lua	Lua	Lua	Action Script
Physic Engine (Bullet)	+	+	-	-	-
Video based features					
Video texturation from file	+	+	-	Without sound	+
Video recording (with sound)	+	+	-	-	Without sound
Screen capture	+	+	+	+	
Image processing	+	+	-	-	-
Web manager	+	+	+	+	Partial*
Inputs					
Mouse	+	+	-	-	+
Keyboard	+	+	Partially supported	-	+
GPS	-	-	+	+	-
Accelerometer	-	-	+	+	-
Multi-touch	-	-	+	+	-
Compass	-	-	+	+	-
Plugin SDK	-	-	-	-	-
Component architecture	Partial	Partial	+	+	-
Barcode plugin**	+	+	+	+	-

(*) partial Web Manager for Flash : many things available directly within Flash
(**) details on supported barcode formats are to be found in D'Fusion AR2 documentation

Figure 2: Cross-platform rendering features

Cross platforms features matrix Reference : D'Fusion Pro	@Home Windows	@Home Mac OS X	Mobile Android	Mobile iPhone	for Adobe Flash Player
MLT Classic	**+**	**+**	**-**	**-**	**-**
Keyframe image format	*	*	-	-	-
Multi Planar Objects	+	+	-	-	-
Multi keyframes AutoInit (Reco)	+	+	-	-	-
3D Object tracking	+	+	-	-	-
Multi 3D Object tracking	+	+	-	-	-
Face detection	+	+	-	-	-
Face tracking	+	+	-	-	-
Exclusion zone	+	+	-	-	-
Pointing detection	+	+	-	-	-
MLT Lite	**+**	**+**	**+**	**+**	**+**
Multi Planar Objects	+	+	+	+	+
Multi keyframes AutoInit (Reco)	+	+	+	+	+
Keyframe image format	*	*	bmp, jpg	bmp, jpg	jpg
3D Object tracking	+	+	+	+	+
Face detection	+	+	+	+	+
Face tracking	+	+	+	+	+
CV Tools					
Motion detection	+	+	-	-	-
Background subtraction	+	+	-	-	-
Move detector	+	+	-	-	+
Driving Tool	+	+	-	-	+
Marker Tracking	+	+	+	+	-

Figure 3: Cross-platform Computer Vision features

3 PROJECT CREATION

This section describes how to ensure the compatibility of your AR applications across a large range of computers (PCs and Macs) when creating your scenarios with D'Fusion Studio. This way, your final application will most likely work properly on the final user's computer, even with low configurations.

For information on how to create AR scenarios, please refer to the relevant documentation [01].

3.1 Project recommendations

3.1.1 Video Capture and Rendering window

Video Capture Size vs. Rendering Window Size:

	Standalone		Web	
	Low	High	Low	High
Video capture size	320*240	640*480	320*240	640*480
Window size	800*600	1024*768	640*480	800*600

Camera selection:

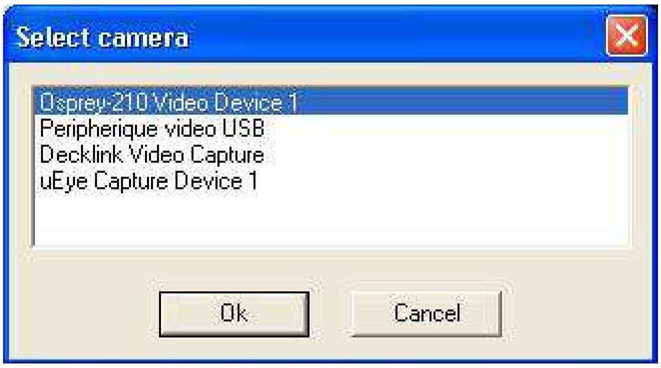
The camera selection functionality is executed only if there are 2 or more cameras recognized as connected to the computer. If there is only one camera recognized, the one-recognized camera is automatically used. Note that even if the user has only one physical web-camera connected, there can be other video input devices recognized (graphical card video input, video acquisition card, virtual cameras, etc.).

To allow the final user to choose the proper camera, the selection camera mode must be enabled from D'Fusion Studio: in the properties of your video capture ("Object Editor" window), check the "Select dialog" option.

You can also open the camera configuration .xml file and add the word "selection" in the parameter:

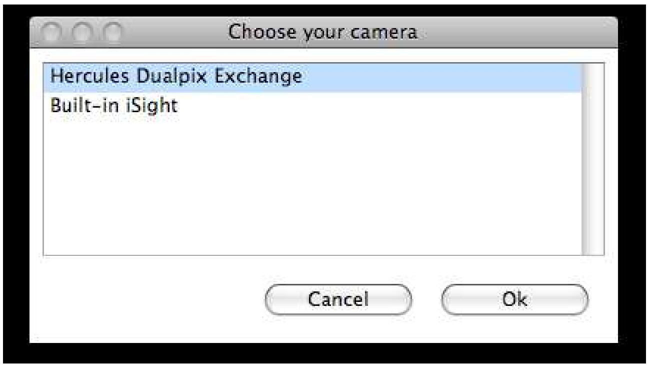
```
<VIDEO_CAPTURE_FRIENDLY_NAME>selection</VIDEO_CAPTURE_FRIENDLY_NAME>
```

With the camera selection option enabled, the application will show a dialog box. The dialog box will have the same look as the OS of the final computer (OS-specific). In a web based project, the dialog box can be customized. More informations is provided in section "5.1.9 Camera selection in detail".



Osprey-210 Video Device 1
Peripherique video USB
Decklink Video Capture
uEye Capture Device 1

Figure 4: Windows camera selection



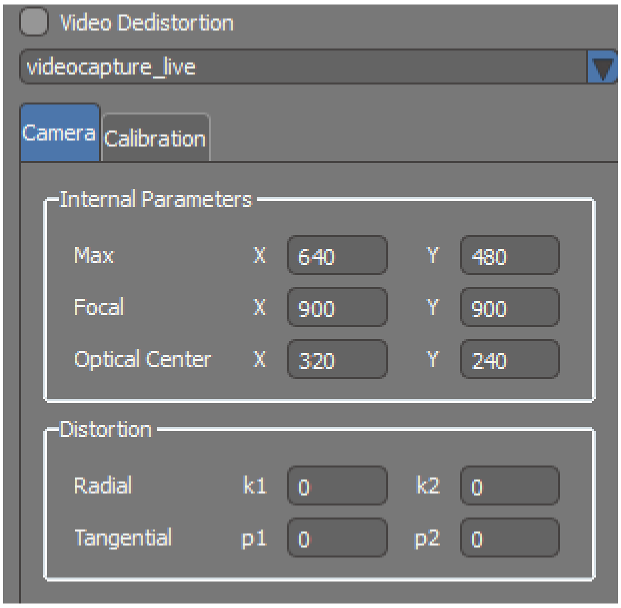
Hercules Dualpix Exchange
Built-in iSight

Figure 5: MacOSX camera selection

3.1.2 Video Capture and Distortion Correction (DEDISTORTION)

The camera used by the final user is usually unknown when developing the application for large scale deployment. It is thus recommended to deactivate the distortion correction process from the video capture.

To deactivate this option, use the following parameters when configuring the camera calibration in D'Fusion Studio:



Internal Parameters

Max	X	640	Y	480
Focal	X	900	Y	900
Optical Center	X	320	Y	240

Distortion

Radial	k1	0	k2	0
Tangential	p1	0	p2	0

Figure 6: Camera Calibration settings

The camera "Max (x,y)" must be the chosen resolution and the "Optical Center (x,y)" must be half of these values.

You can also deactivate the distortion correction process by setting the DEDISTORTION parameter to false in the camera calibration .xml file:

```
<DEDISTORTION>false</DEDISTORTION>
```

3.1.3 Tracking

Here are our recommendations:

- 2 objects tracked at the same time maximum
- 1 planar object with 20 autoinit keyframes maximum (choose one detection by frame for the autoinit mode)
- 300 interest points maximum
- For 3D tracking you need to use ".obj" files.

The tracking initialization phase can be very long. This phase is proportional to the number of tracked objects. To avoid an extended amount of time, it is recommended to split the tracking part of the scenario by making one tracking configuration file for each tracked target, and loading the tracking configuration file when the user has chosen the object to track.

Note that the tracking scenario must be configured in "Classic" mode in D'Fusion Studio "Computer Vision" panel.

3.2 Media recommendations

3.2.1 Rendering generalities

The D'Fusion @Home players are designed to use the platform default renderer.

The OpenGLES 2 renderer is not supportd by the D'Fusion @Home players.

3.2.2 Cross-platform projects

To distinguish media or parameters specific to Windows from media or parameters specific to Mac OS X, you can check the "Alternative platform" option of the media's properties (in D'Fusion Studio's "Object Editor" window), then choose an alternative media for each platform.

3.2.3 Short names for files and paths

The length of file names (character number) must be limited. Some operating systems do not support too long names.

Similarly, the path name must not be too long (so the folder number must also be limited).

Example:

File names like ***Object_3d_that_have_an_excessivly_long_name.mesh*** **are strictly forbidden!**

Path like ***scenario/media/folder_1/folder_2/folder_3*** **are strictly forbidden!**

3.2.4 Anti-aliasing

Anti-aliasing can be activated from the D'Fusion Studio "Project Settings" window.

It is recommended to use a value of 4 for the anti-aliasing (or an even number power of two). If the anti-aliasing value is not supported by the graphics card, the software will automatically decrease the value until a supported value is found.

3.2.5 Lights

It has been noted that all 3D objects were blank on some computers. The issue is that some graphic cards do not handle lights with zero constant attenuation.

The solution to this problem is to avoid using 0 for the "constant" parameter of the light properties (D'Fusion Studio "Object Editor" window).

3.2.6 Textures

Format:

Uncompressed file formats (such as TGA and BMP) are to be avoided, since they are too heavy and will take a long time to download. JPG and PNG formats should be preferred:

- JPG for natural images
- PNG if transparency is needed or if a large part of plain color is used, PNG format is preferred to JPG because it has a better compression quality

Size:

Downscale the textures. This will decrease the loading time and allow the application to run on lower grade computers without graphic cards (like Intel chipset).

We recommend using less than 512*512 pixels size. For example, for a D'Fusion @Home window of 800*600 pixels, if the 3D object takes only 50% of the window, a 256*256 pixels texture is enough.

For materials and textures that do not use mip-maping (mip-map is set to 0), the texture sizes must be power of 2. It means width and height must have one of these values (2, 4, 8, 16, 32, 64, 128, 256, 512, etc.)

By using an adequate texture resolution, you will gain:

- Good visual quality, the texture display well in the final rendering
- Less GPU memory, then a faster rendering
- Smaller file size, which means a faster downloading time

3.2.7 Meshes

Please, read the relevant documentation [06 & 07] and [08 & 09].

Size:

We recommend having less than 100.000 polygons displayed on the screen at the same time. Each individual mesh file must have less than 63.000 polygons. The number of polygons affects the amount of time it takes to download an application. Reducing the number of polygons in a scene is a good way to keep the download time lower.

Re-use shared part:

Many 3d meshes have identical meshes or sub meshes, but have different textures. We recommended using one mesh for multiple 3d object instances. This will decrease downloading time and improve 3d rendering.

D'Fusion Exporter for Maya: "Public" mode is the only mode supported on all office PC with Intel graphic board, even if they are powerful.

3.2.8 Sound

We recommend using the **".ogg"** format because the sound size will be smaller and it will be compatible both with Windows and MacOsX.

"Mono" channel sounds are preferred; dual channel double the sound file size.

For sounds that must be recorded during video recording, they must be Mono, 16 bits, 44100 Hz and the parameter **"streaming"** must be set to **"true"**.

3.2.9 Video

We recommend either embedding the video with your application or embedding the video with one of the downloaded data folders used by your application. Please read the size recommendations below.

You can also play a video hosted on another server by providing the address to the video. In such case, the video is read in the distant server and played in your application, which can result in less smoothness than with standard streaming. Constraints with this method include:

- The address must be static, a dynamic link will not work
- The video can't be encrypted, or otherwise it won't be read
- If the connection to the server is lost, the reading will not pick up where stopped

In both cases, please check that the format is compatible with D'Fusion, as explained below.

Size:

- Avoid excessively long video; video less than 1 minute is preferred.
- Choose the size according to the D'Fusion window: 640*480 or 320*240.
- Keep in mind that the video size will directly impact downloading time.

Format:

- Recommendations according to the OS:
  - Windows - Format: .avi, Codec: MJPEG or Format: .wmv, Codec: WMV9
  - MacOsX - Format: **.mov**, Codec: **Mpeg4**

3.3 Effects recommendations

3.3.1 Particles

Please note that some parameters are not available on Mac Os X. List of supported parameters:

- Emitter Type 'Point'
- Emitter Type 'Box'
- Emitter Type 'Ellipsoid'
- Emitter Type 'Cylinder'
- Emitter Type 'Ring'
- Emitter Type 'HollowEllipsoid'
- Affector Type 'LinearForce'
- Affector Type 'ColourFader'
- Affector Type 'ColourFader2'
- Affector Type 'ColourImage'
- Affector Type 'ColourInterpolator'
- Affector Type 'Scaler'
- Affector Type 'Rotator'
- Affector Type 'DirectionRandomiser'
- Affector Type 'DeflectorPlane'

3.3.2 Shadows

Shadows can be added to the 3D scene. Although it impacts performance, shadows add a lot of realism to 3D rendering. To minimize the impact of shadows on rendering performances, follow these recommendations:

- Shadows must be activated with only one light source.
- The light source must be a spot light type, not a directional or point light type.
- The maximum resolution for a shadow texture is 1024x1024. 512 is generally fine.
- Shadows must be activated on moving objects only. For static objects, pre-calculated shadows must be used.

3.3.3 Environment mapping

If an environment mapping texture is applied on an object, the material must not have the argument "scene_blend alpha_blend" or shadow casting **must be disabled** on the object. Otherwise a crash is generated.

3.3.4 Shaders

If a material uses a shader, a second rendering technique must be added to the material to handle non compatible graphic cards. If the backup rendering technique is not in the material, the material will be blank or you will have a crash.

If the @Home application is being used on a large user platform, the CG language is preferred to other shading language. HLSL is for DirectX and will not work on Mac OS X platform.

A list of existing profiles on graphic cards can be found at: http://developer.nvidia.com/object/cg_profiles.html

More generally, the use of shaders is not recommended for @Home deployment.

Moreover, please note that the D'Fusion tiShaders have not been designed and are not recommended for the D'Fusion @Home Deployment. For any further information on the DFusion tiShaders please refer to the document "DFusion tiShaders – User Guide".

To optimize the rendering, the "Default Renderer" rasterizer, a DFusion project setting, should be set. This will enable the use of openGL for macOSX, and directX3D for windows.

3.4 Mouse use

The following table describes the cross-platform mouse button availability for D'Fusion @Home Web applications. It will help you define the optimum use of the mouse in your @Home Web scenarios for large platform compatibility.

Mouse	@Home Windows Internet Explorer	@Home Windows Firefox	@Home Windows Chrome	@Home Mac OS X ≥ 10.6 Safari	@Home Mac OS X < 10.6 Safari	@Home Mac OS X Firefox
Left button	+ *	+	+	+	+	+
Right button	+ *	+	+	+	-	-
Wheel	+ *	+	+	+	-	-

Figure 7: Cross-platform mouse button Position and Click availability

(*) Using Internet Explorer in Protected Mode (under Windows OS platform > XP like Vista, Seven …), the input mouse is limited to 3 application reloads in the same web browser instance. No problem in using other web browsers or IE in non Protected Mode under Windows.

4 TEST AND PROJECT VALIDATION

Once your project is ready, you may want to test it on several computers and in conditions similar to the final release (encryption settings, required files, packaging) before proceeding to final deployment.

You can do your tests using the Test Software Key (.dfk file) that is automatically generated when your project is exported.

This key will allow you to play-test your final AR application with the suitable player (D'Fusion @Home Desktop Player for @Home Desktop applications or D'Fusion @Home Web Player for @Home Web applications) and in final-like conditions.

If you are a Pro user, you can also use the tiProtectorAR2 command line tool to generate compatible test keys (or full scenario processing). See section 6.5 for more information on how to use this tool.

4.1 @Home Desktop applications

You can test your @Home Desktop application using the D'Fusion @Home Desktop Player already installed in your computer.

The D'Fusion @Home Desktop Player is installed with D'Fusion Studio. This player can be found in the D'Fusion Studio "Deployment" installation directory and it will allow you to test your @Home application on a Windows PC. @Home Desktop applications for Mac can be tested on Windows before proceeding to final deployment, described in section "5.2.2 Standalone packaging on MacOSX platform".

To launch the player and test your @Home Desktop application on your PC, use the following command line:

```
            %HomePlayerDirectory%\DFusionHomePlayer.exe –f
YourExportedProjectDirectory\YourExportedProject.dpd
```

In Test mode (i.e. a Test Software Key is associated with the exported scenario), the player displays a tattoo:

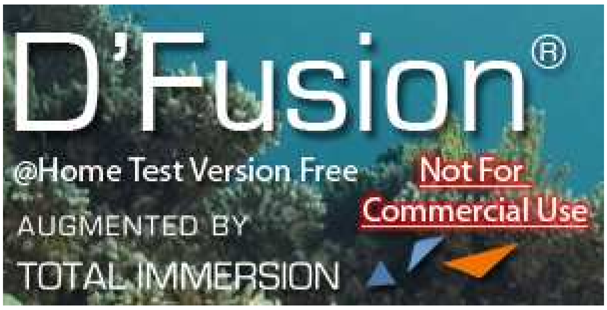


Figure 8: @Home Desktop test version tattoo

4.2 @Home Web applications

D'Fusion @Home Web applications can be tested online (using a web server) or locally (on a local computer).

If your scenario has some interactions with a web page, it is recommended that you test your application with all the interactions and callbacks such as JavaScript functions.

The following sections describe how to test your @Home Web application.

4.2.1 Download D'Fusion @Home Web

Here are the instructions to start using D'Fusion @Home Web:

- Download D'Fusion @Home Web from our website. This package contains the players for Windows and Mac OS X, the default JavaScript files, and the web browser plug-ins for @Home Web AR applications.
- Extract the package in your project folder.

  Note: If you have more than one project requiring D'Fusion @Home Web, download the package and extract it once for each project in each project folder (see the folder structure below).
- Once you have D'Fusion @Home Web on your computer, export your project with D'Fusion Studio.
- Ideally, you should organize your projects with the following directory structure:

```
+ MyProject              - A folder containing a D'Fusion project
  + DFusionProject       - Project folder: contains all D'Fusion Studio
                           project files (project.dpd, media, etc).
  + ExportedProject      - Exported project folder: contains all exported
                           files (project.dpd, media, test keys, etc).
  + DFusion@HomeWeb      - Extracted D'Fusion @Home Web package
    + local
    + online
    + tools
+ AnotherProject         - A folder containing another D'Fusion project

...
```

Once your project is exported, you can start testing your @Home Web application. We advise you to test your application locally before proceeding to an online test.

Note: the above directory structure will allow you to have your exported project folder next to your D'Fusion Studio project folder, with each targeted deployment platform readily visible.

4.2.2 Local test

To test your @Home Web application locally:

- Copy the contents of your exported project folder into the "scenario_test" folder available in the "D'Fusion @Home Web" > "local" folder:

```
+ DFusion@HomeWeb
  + local
    + scenario_test
```

  By default, "sceneario_test" contains the minimum files required to test your @Home Web application locally.

- Launch "sampleAR2_test.html" in a web browser.

  If your exported project is not named "project.dpd", you will have to edit line 30 of "sampleAR2_Test.html" ("`scenario_test/project.dpd`") for your application to launch properly. See also "4.2.4 HTML integration" below.

4.2.3 Test web server

Details on how to proceed to Final deployment are provided in section 5.1. Test deployment is similar, except that you don't need to use the PrepareFolderForUpdate tool described in section 5.1.2.

You can test your @Home Web application online using an online web server:

- Prepare your exported project folder for download using the PrepareFolderForUpdate tool. This tool generates a package ready for download, including the file ".update.xml", which lists all the files that must be downloaded. Refer to section "5.1.2 PrepareFolderForUpdate tool" for more information on how to use this tool.
- Copy the contents of this folder into the "scenario_test" folder available in the "D'Fusion @Home Web" > "online" folder:

```
+ DFusion@HomeWeb
  + online
    + scenario_test
```

  By default, "sceneario_test" contains the minimum files required to test your @Home Web application online.

- Configure your FTP tool so as to ensure that copying is done in binary mode and that hidden files are also downloaded. More information on FTP is available in section 5.1.4.
- Upload the content of your "online" folder to your online web server.
- Open a web browser and connect to "sampleAR2_Test.html" on your server.

  If your exported project is not named "project.dpd", you will have to edit line 30 of "sampleAR2_Test.html" ("`scenario_test/project.dpd`") for your application to launch properly. See also "4.2.4 HTML integration" below.

Your D'Fusion @Home Web application can now be tested online.

When your AR scenario has to dynamically download data from a web server, you will also need to generate Test Key(s) for the data to download:

- Use the tiProtectorAR2 tool on the folder containing the data to download (and dynamically use). The tiProtectorTool will *export* this data, i.e. it will generate a Test Software Key for this data and encrypt it if necessary. Please, read chapter 6.5.
- Prepare this folder as a package to download using the PrepareFolderForUpdate tool. Please refer to section "5.1.2 PrepareFolderForUpdate tool" for more information. Then, you can follow the steps described above.

4.2.4 HTML integration

You should at least include the following code inside the HTML <body> tag of your *test.html* file:

```
<body>
        <div id="InteractionCell”></div>
        <div id="plugin_dfusion”>
        <script language="JavaScript”>
                var player = "";
                if (tiMacOSXSupported()==false) { // Windows version
                        player = "DFusionWebPlayer/DFusionWebPlayer”;
                } else {
                        player = "DFusionWebPlayerMacOSX/DFusionWebPlayer”;
                }

                tiGenerateDFusionWebHTML(
                        "DFusionApplication”, // project specific identifier
                        800, 600, // x and y
                        player, // test player to be used
                        "my_scenario/project.dpd”, // exported scenario to be used
                        "ChooseCamera”, // JS function invoked for HTML camera selection
                        "PersonalizedAction”, // JS function invoked when the application is ready
                        "true”, // camera selection mode: if true = selection is made via Web-Page
                        "SplashConf/splash.ssc” // customized splash screen file
                );
        </script>
        </div>
</body>
```

Make sure that the names and paths in the web page are correct.

You will find information on Final web integration in section 5.1. Test web integration is similar, except that the augmented experience is tattooed.

4.2.5 Web server directory structure

On the web server, the most commonly used directory structure is:

```
+ MyProject (test content folder)
  - index.html (main web page)
  - other files (html, image, ...)
  + js
    - some JavaScript files …
  + plug-in
    - D'Fusion navigator plug-in files
  + DFusionWebPlayerTest
    + resources
      - player resource files (including default splash screen)
    - .update.xml
    - DFusionWebPlayer.dll
    - splash.zip
    - other dlls ...
  + DFusionWebPlayerMacOSX
    - .update.xml
    + DFusionWebPlayer.bundle
  + scenario
    - .update.xml
    - project.dpd.dfk
    - project.dpd
    - other files (3D, audio, video, configuration files)
```

Note especially that there are hidden files named ".update.xml" that must be present on the online server for the automatic download functionality to work properly.

4.3 Final validation

When deployment test is completed, you'll have to send your Test Software Key to Total Immersion, who will process the key according to your contract and generate a final deployment key called Final Software Key.

The Final Software Key will then be sent back to you. This key is to be used in the exact same way as the test one: replace your test keys with your final ones in your application's export folder and **re-apply steps for scenario packaging (such as "PrepareFolderForUpdate" for web applications)**, then use it like in deployment test.

5 FINAL DEPLOYMENT

5.1 Web integration

5.1.1 D'Fusion @Home Web application structure

There are 3 main parts that compose the D'Fusion @Home Web solution:

- Plug-in – the application that allows integration into the browser
- Player – the 3D engine
- Scenario – the 3D media and scripting files needed for the AR experience

The following figure illustrates the interconnection between these 3 parts and the supported web browsers:

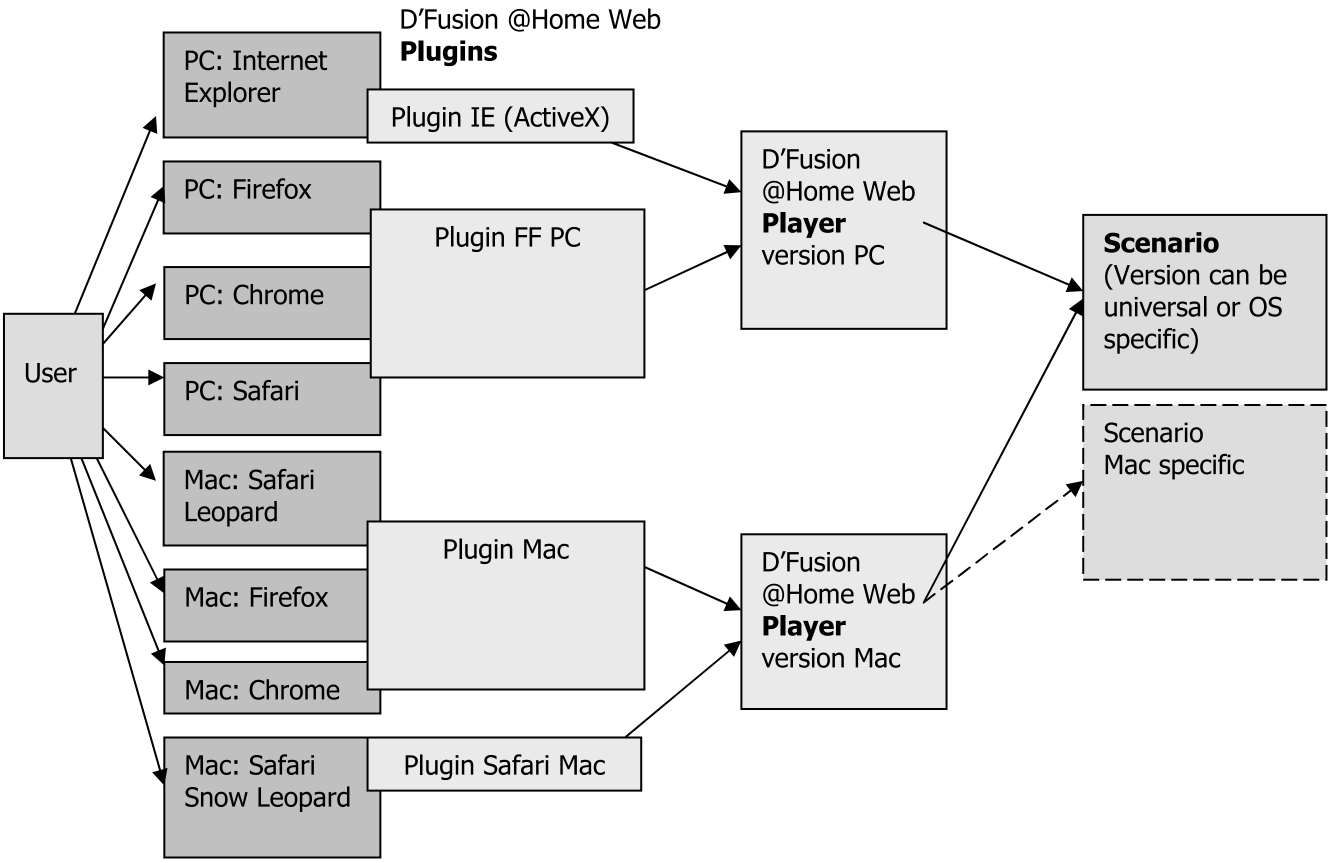


Figure 9: D'Fusion @Home Web architecture

The exact usage of these components is:

D'Fusion @Home Web Plug-in = project-independent, platform-specific and browser-specific

- An application that provides a "glue" to the web-browser executed by the final user
- It has its own installer: the installation is made in the same way as Adobe's Flash: It is an on-usage check. If the application is already installed the application will be installed automatically
- Web Plug-in has a form of ActiveX for Internet Explorer, Mozilla Plug-in for Firefox
- It allows to launch "Players" and handles the interface between the "Player" and the Web page itself
- The plug-in is generic for all our @Home Web products. So, the actual 3D computation is not contained in the plug-in, but in the "Player"

D'Fusion @Home Web Player = project-independent, platform-specific

- 3D engine of the application
- Binary package (compiled code)
- Binaries are specific to a given platform: Win/Mac. So there are 2 binaries (Win and Mac)
- Presented in the form of a .dll (Win) / bundle (Mac) collection
- For a given platform (either Win or Mac), it is browser-independent
- Allows the loading of one or more scenarios per project (see "Scenario" below)
- One player execution = one scenario is interpreted

Scenario = 3D media and configuration = project-specific, either platform-specific or platform-independent

- A collection of configuration files, 3D medias, sounds and videos (if any)
- It can be platform-independent (i.e. the same configuration for Win/Mac) or platform-specific (if there is some media that is platform-specific, especially video = e.g. WMV for Win, MOV for Mac)

5.1.2 PrepareFolderForUpdate tool

The PrepareFolderForUpdate tool is installed with D'Fusion Studio, in the folder installation directory > Deployment > Tools.

This is a command line application that allows you to prepare the scenario data for download, for @Home Web projects. You can optionally compress the scenario project files in order to decrease the download time.

The generated .update.xml indexes all the compressed data that must be downloaded.

Command parameters are:

Option	Name	Description
-i	input folder name	Set the folder where the program will browse to find media to download
-o	output folder name	Set the folder where the program will put data
-c	Compress option	All files from input folder will be compressed before being copied to output folder
-ie	extension to ignore	concerned files won't be in the output folder
-if	folder to ignore	concerned folders won't be in the output folder
-iec	extension to ignore for compression	concerned files will be copied without compression

For the "ie", "if" and "iec" options, a list can be provided with commas used for separations.

For the input folder, keep in mind that a path with a space must be enclosed with quotes.

Example:

```
PrepareFolderForupdate.exe –i "scenario" –o "onlineScenario" –c –iec "zip,wmv,mp4,jpg,png"
```

5.1.3 Application download, initialization & execution

For a user that has never used any of D'Fusion products:

- First the plug-in needs to be installed. The plug-in presence / non-presence is checked automatically. If not present, the plug-in installer is downloaded and launched automatically. A special case for Firefox: Firefox needs to be closed during installation (please note this is a Firefox issue and that Adobe Flash Installer is doing the same.)
- Once the plug-in is installed, the player is downloaded automatically
- Once the player download is finished, the plug-in launches the player
- The player downloads the scenario to be executed (one scenario at a time)
- The player initializes the 3D scenario – i.e. possible creation of necessary data-structures
- The player initiates camera selection form (via JavaScript) but only if there are two or more cameras
- The player launches the 3D engine and ensures the JavaScript communication (input/output)

Any future launch will be faster. If the player and/or media have not changed, then no additional data download is necessary and data-structures already created are reused:

- The plug-in launches the player
- The player initializes the 3D scenario
- The player initiates camera selection form (via JavaScript) but only if there are two or more cameras
- The player launches the 3D engine and ensures the JavaScript communication (input/output)

5.1.4 Web server directory structure

The most common directory structure on the web-server for D'Fusion @Home Web applications is listed below:

```
+ MyProject
  - index.html (main web page)
  - other files (html, image, ...)
  + js
    - javascript_browser_sniffer.js
    - totalimmersion_dfusion_web.js
    - totalimmersion_dfusion_web_config.js
    - totalimmersion_dfusion_web_custom.js
  + plugin
    - DFusionHomeWebPlugin.Installer.dmg
    - DFusionHomeWebPlugin.Installer.exe
    - totalimmersion_dfusion_web_plugin.js
    - totalimmersion_dfusion_web_pluginMac.js
  + DFusionWebPlayer
    + resources
      - player resource files (including default splash screen)
    - .update.xml
    - DFusionWebPlayer.dll
    - splash.zip
    - other dlls ...
  + DFusionWebPlayerMacOSX
    - .update.xml
    + DFusionWebPlayer.bundle
  + scenario
    - .update.xml
    - project.dpd
    - project.dpd.dfk (final key delivered by TI)
    - other files (3D, audio, video, configuration files)
  + scenario-Mac (optional)
    - .update.xml
    - project-Mac.dpd
    - project-Mac.dpd.dfk (final key delivered by TI)

    - other files (3D, audio, video, configuration files)
```

Note: there are hidden files named ".update.xml" that must to be present on the server and are used for the automatic download functionality.

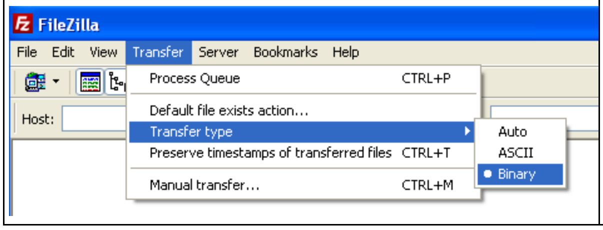
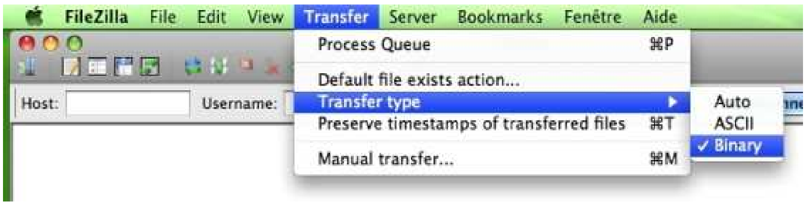
Note: There are two versions of the D'Fusion @Home Web Plugin Installer for Windows, the "DFusionHomeWebPlugin.Installer.exe" and the "DFusionHomeWebPlugin.InstallerFull.exe". The first version detects if the Package redistribuable Microsoft Visual C++ 2010 (x86) is already installed on the target computer and, if not, downloads it from the Microsoft web site and installs it. The second version integrates the Package redistribuable Microsoft Visual C++ 2010 (x86) so doesn't download it if needed. By default, the javascript sample files given for web integration use the "non-full" version "DFusionHomeWebPlugin.Installer.exe".

5.1.5 Putting files onto the web server

To prevent any future problems, be very careful when you are uploading your files onto your web server:

All file transfers need to be binary, even if the file extension corresponds to a textual file.

Thus, be sure that the way you are actually putting the files on the web server is correctly performing a binary transfer on all files. Typically, if you are using an FTP client, be sure your communication is correctly configured in a "Binary" mode, avoiding "ASCII" and "Automatic" (Filezilla example here):

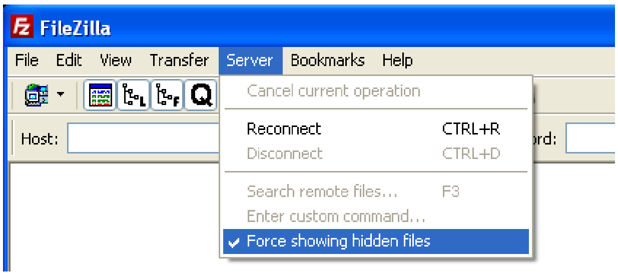
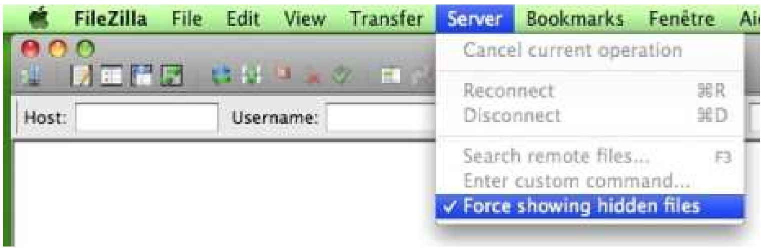
BINARY TRANSFER SETUP IN FTP CLIENTS	
Filezilla - Windows version	Filezilla – Mac OS X version
	

FileZilla
File Edit View Transfer Server Bookmarks Help
Process Queue CTRL+P
Default file exists action...
Transfer type
Preserve timestamps of transferred files CTRL+T
Manual transfer... CTRL+M
Auto
ASCII
Binary
Host:

FileZilla File Edit View Transfer Server Bookmarks Fenêtre Aide
Process Queue ⌘P
Default file exists action...
Transfer type
Preserve timestamps of transferred files ⌘T
Manual transfer... ⌘M
Auto
ASCII
✓ Binary
Host:
Username:

Figure 10: Filezilla transfer mode

Also, do not forget that there are hidden files to be transferred (those beginning by a dot ".", especially the ".update.xml".) To be able to see such files, be sure that your ftp client (or any other directory browser) is able to display such files (locally and remotely.) If not, you may encounter problems with FTP clients when trying to erase directories that seem to be empty but still contain the ".update.xml" file.

FORCED SHOWING OF HIDDEN FILES IN FTP CLIENTS	
Filezilla - Windows version	Filezilla – Mac OS X version
	

FileZilla
File Edit View Transfer Server Bookmarks Help
Cancel current operation
Reconnect CTRL+R
Disconnect CTRL+D
Search remote files... F3
Enter custom command...
✓ Force showing hidden files
Host:

FileZilla File Edit View Transfer Server Bookmarks Fenêtre Ai
Cancel current operation
Reconnect ⌘R
Disconnect ⌘D
Search remote files... F3
Enter custom command...
✓ Force showing hidden files
Host:
Username:

For instance, in the Filezilla FTP client, you need to have the option "Force showing hidden files" activated:

Figure 11: Filezilla hidden files mode

We do not recommend using Windows Explorer's FTP capability for web server uploads and configurations as this specific client does not provide full configuration functionality (e.g. does not permit to show hidden files).

5.1.6 Required web server configuration for download

The binary mode for download has to be ensured by the web server. This must be configured for all files, even if their extension corresponds to a textual file.

Files have to be accessible for download.

To prevent any future problems, please ensure that this is supported and funcional on your hosting web server.

Note that these points are mandatory and if not ensured, the D’Fusion @Home Web application will not function at all, because either no files could be found on the server by the D’Fusion @Home Web application, or, if found, corrupted files may have been downloaded.

If your server is managed by IIS, please refer to the section “Issues to be prevented with IIS” in section 8 Troubleshooting”

5.1.7 HTML Integration

The prerequisite for the HTML integration is the inclusion of the following standardized JavaScript files into the HTML’s <head> tag. The project specific JavaScript functions (PersonalizedActions here) are to be defined in the same place (the **PersonalizedActions** function here):

```
<head>
        <!-- some project specific HTML’s head data here -->
        <script src="./Plugin/totalimmersion_dfusion_web_plugin.js”></script>
        <script src="./Plugin/totalimmersion_dfusion_web_pluginMac.js”></script>
        <script src="./js/javascript_browser_sniffer.js”></script>
        <script src="./js/totalimmersion_dfusion_web_config.js”></script>
        <script src="./js/totalimmersion_dfusion_web.js”></script>
        <script src="./js/totalimmersion_dfusion_web_custom.js”></script>
</head>
```

In a default, minimal way, the necessary code to be included into the HTML <body> tag is:

```
<body>
        <div id="InteractionCell”></div>
        <div id="plugin_dfusion”>
        <script language="JavaScript”>
                var player = "";
                if (tiMacOSXSupported()==false) { // Windows version
                        player = "DFusionWebPlayer/DFusionWebPlayer”;
                } else {// MacOSX version
                        player = "DFusionWebPlayerMacOSX /DFusionWebPlayer”;
                }

                tiGenerateDFusionWebHTML(
                        "DFusionApplication”, // project specific identifier
                        800, 600, // x and y
                        player, // player to be used
                        "my_scenario/project.dpd”, // scenario to be used
                        "ChooseCamera”, // JS function invoked for HTML camera selection
                        "ReadyToStart”, // JS function invoked when the application is ready
                        "true”, // camera selection mode: if true = selection is made via Web-Page
                        "SplashConf/splash.ssc” // customized splash screen file (optional)
                );
        </script>
        </div>
</body>
```

Please note that the entire code mentioned above must be incorporated into the web page, especially the **InteractionCell** used for the default HTML camera selection functionality. In addition, please note that the **focus** function call is necessary for the application to function correctly when incorporated within an iFrame on Firefox-like browsers.

The inclusion of the application itself in the web-project is done within the HTML <body> tag by invoking the **tiGenerateDFusionWebHTML** function, called with project-specific parameters. This function takes 9 parameters:

1. Name of the plug-in object (DFusionApplication): this name will be used if you want to interact with the plug-in/player using JavaScript.
2. Width of the window used by the plug-in. This value should be equal to the rendering dimensions declared by the project-specific scenario.
3. Height of the window used by the plug-in. This value should be equal to the rendering dimensions declared by the project-specific scenario.
4. Path to the absolute or relative path to the player file (dll under Windows and bundle under Mac OS X, but you can avoid the extension.)
5. Path to the scenario: absolute or relative path to the configuration file
6. Callback to select the camera (CameraOptions function in the listing below): function called if more the one capture device is present on the end user's computer. The goal of this function is to display a list of devices and let the user choose the correct one. If you want a different behavior than the default, take the body of this function from the totalimmersion_dfusion_web_custom.js file, create a new function and modify the behavior to meet the exact project requirements.
7. Callback is invoked when the player has finished initializing. (ReadyToStart function): function called when the player starts the rendering. It can be used to modify the web page when the player starts. For instance, to make a menu appear with HTML buttons/elements to control the @Home Web application via JavaScript.
8. Camera selection Boolean switch: if true, the camera selection is made via Web-Page and JS mechanism, If false, the application's internal dialog box (look is OS specific) will be used
9. Path to a project-specific splash screen configuration. If not specified, a default version of the splash screen is used.

This function generates an in-place HTML code containing the correct links to all the files required to run your @Home Web application (plug-in, player and scenario) stored on the distant side (web-server). This gives the final web integrator the possibility to keep the code of the main page minimal, while retaining the ease of future modification of one or several parameters.

5.1.8 Communication with the web page using JavaScript

The D'Fusion @Home Web application has the capability of communicating with a web page using JavaScript:

- output: capable to invoke a JavaScript function with or without arguments
- input: it is possible to pass arguments from the JavaScript to the application

A default communication protocol is provided ensuring a basic communication. If this is not sufficient, the default protocol can be modified and adjusted to meet the exact needs of the Web project. This is done by appending input/output invocations, enriching the basic communication protocol.

5.1.8.1 Output to JavaScript

The D'Fusion @Home Web application is capable of invoking any prior known JS function via its name (also called JS call-backs) with or without arguments. In the case of JS, all arguments are strings.

If the callback function does not exist in the actual loaded web-page document, the invocation is skipped and nothing happens.

5.1.8.2 Input from JavaScript

In the *.js files provided with the web application, you can find an "ExecuteCommand(command)" JavaScript function. This function binds directly the "command" argument to the web plug-in. This single argument has to be a string formatted in the following way:

```
function_name 'arg0'arg1'arg2'...
```

The function_name refers either to:

- The pre-defined D'Fusion @Home Web command (not case sensitive):

  CAMSELECTED – used for camera selection

- Or, in any other case, the function is supposed to be project-specific and is dispatched to the applicable layer configurable by the scenario.

Arguments can be omitted or present. If present, an argument's definition is surrounded by quotation marks (in the beginning and at the end). If more than one argument is passed, the arguments must be separated by an extra set of quotation marks.

Example of JS function that will invoke keypresses within the D'Fusion @Home Web application:

```
function MakeKeystrokeActions(){
                for (var i = 0; i < arguments.length; i++) {
                        ExecuteCommand("PLAYWITHKEY '"+arguments[i]+"'");
        }
}
```

PLAYWITHKEY is the name of a pre-defined D'Fusion @Home Web command that will ensure a SW keystroke.

This function can be called as follows:

```
MakeKeystrokeActions( "y", "a", "m" );
```

This call will invoke the ExecuteCommand function three times and ensure 3 SW keystrokes in this way:

```
ExecuteCommand("PLAYWITHKEY 'y'");
ExecuteCommand("PLAYWITHKEY 'a'");
ExecuteCommand("PLAYWITHKEY 'm'");
```

5.1.8.3 Signaling error with JavaScript

In the case of any irregular behavior in the D'Fusion @Home Web Plug-in / Player / Scenario execution, a specific JavaScript function is called during the experience so that the web-page can monitor the actual status.

This function is defined in the file totalimmersion_dfusion_web_custom.js. Modify it so it feet your needs

```
function DFUSION_EVENT_Error(status)
{
        //add your code to display information
}
```

Most of these errors are used for internal debugging. The only error that will be useful on a final (release) version is the TI_EETH_LOAD_VIDCAP_FAILED, received when the webcam can't be used.

Error codes		
Status	**TI internal identification**	**Description**
1	TI_EETH_INIT_PLUGIN_FAILED	Problem with plug-in initialization
2	TI_EETH_DL_SPLASH_FAILED	Problem with splash screen download
3	TI_EETH_LOAD_SPLASH_FAILED	Problem when charging splash screen
4	TI_EETH_DL_PLAYER_FAILED	Problem with player download
5	TI_EETH_LOAD_PLAYER_FAILED	Problem when charging player
6	TI_EETH_INIT_PLAYER_FAILED	Problem with the player initialization
7	TI_EETH_PLAYER_FAILED	Problem with the player execution
8	TI_EETH_DL_SCENARIO_FAILED	Problem with scenario download
9	TI_EETH_LOAD_SCENARIO_FAILED	Problem when charging scenario
10	TI_EETH_LOAD_VIDCAP_FAILED	No webcam available or the webcam is already used or not properly installed

5.1.9 Camera selection in detail

An HTML camera selection mode is supported and provides enhanced "web-page" integration.

In this case, the D'Fusion Web application is communicates with the JS and it is the HTML that ensures the visual representation of the camera selection dialog box. Once selected, the JS passes the chosen camera to the D'Fusion Web application.

The actual mode is chosen via the **tiGenerateDFusionWebHTML** function argument, as described previously in this document.

The HTML-based camera selection is done via JS calls and via a web-page element that provides selection functionality (ComboBox). The actual sequencing is as follows:

- D'Fusion @Home Web Player executes a JS function and passes a list of cameras that were actually recognized on the PC. The name and contents of this JS function are configurable and can be completely project-specific, as already mentioned above. The camera-selection function name is passed to the player when generating the HTML via the **tiGenerateDFusionWebHTML**; note the "**SelectCamera_function**" argument and the camera selection switch that is on "**true**":

```
tiGenerateDFusionWebHTML(
        "DFusionApplication
        800, 600,
        "DFusionWebPlayer/DFusionWebPlayer"
        "my_scenario_pc/project.dpd",
        "SelectCamera_function", // JS function invoked for HTML camera selection
        "ReadyToStart",
        "true", // camera selection mode: if true = selection is made via Web-Page
        "SplashConf/splash.ssc" // customized splash screen file
)
```

- Web page JS function creates a selection dialog.
- When selecting a camera, the D'Fusion @Home Web Player execution is interrupted. The application is will wait until the user selects a camera.
- User chooses a camera. Web page JS intercepts that the user has finished the selection.
- Actual selection has to be signaled by the JS to the D'Fusion @Home Web Player via the following (note the CamSelected standardized command and note that the camname has to be one of cameras identifiers that the D'Fusion @Home Web Player has passed to the JS in the cameras list):

```
ExecuteCommand("CamSelected '"+camname+'""');
```

This mechanism is really flexible. D'Fusion @Home Web does not force in any specific way how the camera selection is to be made. It can be a Flash application that handles all that camera selection choices.

From the D'Fusion @Home Web application point of view there is only:

- An output via JS function transmitting to the web-page with a list of cameras
- A waiting loop –will wait until the web page executes and input JS function with "CamSelected" command + camera name as an argument.

The typical (default) JS code for the camera selection is to be found in the following file:

\js\totalimmersion_dfusion_web_custom.js

The function, once invoked, is injecting an HTML code containing a ComboBox with the camera list and a button so that the user can validate the selection into the following div:

```
            <div id="InteractionCell"></div>
```

This div has to exist within the document (as was mentioned previously), otherwise, the HTML page will not display any camera selection and the D'Fusion @Home Web application will be blocked in an infinite waiting loop for the user's selection that never arrives (so, in a dead-lock).

```
function ChooseCamera()
{
        var SelectFormHtml = "";
        SelectFormHtml += "<FORM name=|"ChooseCamera|">";
        SelectFormHtml += "  <SELECT  NAME=|"CAMERAS|">";
        SelectFormHtml += "  </SELECT>";
        SelectFormHtml += "  <input type=button value=|"camera selected|"onclick="
        SelectFormHtml += "CamSelected(this.form.CAMERAS.options"
        SelectFormHtml += "[this.form.CAMERAS.selectedIndex]. value)>";
        SelectFormHtml += "</FORM>";

        document.getElementById("InteractionCell").innerHTML = SelectFormHtml;
        for (var i = 0; i < arguments.length; i++)
        {
          document.forms['ChooseCamera'].CAMERAS.options[i] = new
          Option(arguments[i],arguments[i]);
        }
}

function CamSelected(camname)
{
        document.getElementById("InteractionCell").innerHTML = "";
        ExecuteCommand("CamSelected '"+camname+"'");
}
```

Note that this is only the default behavior and the application integrator can entirely change the contents of this function, replacing it by a project-specific one. In such a case the "InteractionCell" existence or name is entirely dependent on the project specific web-page construction.

5.1.10 JavaScript customization

In the file **totalimmersion_dfusion_web_custom.js**, you can find a set of function you can customize to meet your needs.

- **tiDFusionWebNotSupported**: This function is called by **tiGenerateDFusionWebHTML()** if the system is not supported. Use this function to display an error message to the end user. **The name of this function must not be changed.**
- **tiDFusionWebCanFindExecuteCommand**: This function is called by **ExecuteCommand()** if the function cannot be called on the plug-in. Use this function to display an error message to the end user. **The name of this function must not be changed.**
- **tiInstallationRequieredForGecko**: This function is called by **tiGenerateDFusionWebHTML()** if the plug-in need to be (re)installed. This function is not used under Internet Explorer but by all other browser (Firefox ...). Uses this function to display information to the end user so it call download the plug-in setup. **The name of this function must not be changed.**
- **tiDFusionPluginNeedToBeUpdated**: This function is only called under Mozilla Firefox MacOSX. Our plug-in version detection does not work correctly on this configuration. So the version is checked while the plug-in is running. The plug-in *ask* the web page which version is required. If the plug-in version is older, it stops and calls this function (defined by **g_NeedToUpdateFunc** in **totalimmersion_dfusion_web.js**). Then you are responsible to do something (change the webpage, remove the plug-in by updated the content of the current web page ...) and then inform the end user it will need to update the plug-in (like what is done in **tiInstallationRequieredForGecko()**). If you change the name of this function update the value of g_**NeedToUpdateFunc in** totalimmersion_dfusion_web.js.
- **ChooseCamera**: Function called by the plug-in when a camera selection is required. The name of this function is passed to the plug-in through **tiGenerateDFusionWebHTML().**This function receive an array of string as unique parameter. Each string is a camera name. To select a camera look at the CamSelected function.
- **ReadyToStart**: Function called by the plug-in when the scenario is starting. The name of this function is passed to the plug-in through **tiGenerateDFusionWebHTML()**. This function does not receive any parameter.

In the file totalimmersion_dfusion_web_plugin.js, you can find a set of global vars to meet you special preferences:

- **g_tiInstallerName**: this variable references the DFusion @Home Web plugin installer to use. By default, the version is **DFusionHomeWebPlugIn.Installer.exe**. This is the non full version that doesn't include the Package redistribuable Microsoft Visual C++ 2010 (x86), and downloads it (and installs it) only if necessary. You can choose at your convenience to use the **DFusionHomeWebPlugIn.InstallerFull.exe** instead**.**

5.2 Standalone version packaging

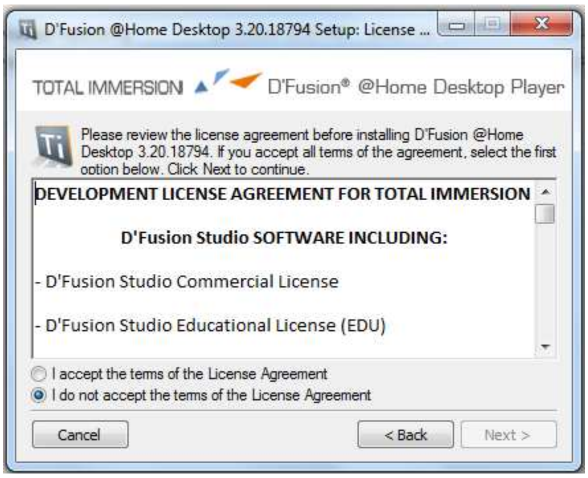
5.2.1 Standalone packaging on Windows platform

5.2.1.1 Installer and license agreement

Many installer packages are available to create the installation setup.

We recommend NSIS.

When creating the installer, the license must be included. Third-party licenses must also be included.



D'Fusion @Home Desktop 3.20.18794 Setup: License ...

TOTAL IMMERSION D'Fusion® @Home Desktop Player

Please review the license agreement before installing D'Fusion @Home Desktop 3.20.18794. If you accept all terms of the agreement, select the first option below. Click Next to continue.

DEVELOPMENT LICENSE AGREEMENT FOR TOTAL IMMERSION

D'Fusion Studio SOFTWARE INCLUDING:

- D'Fusion Studio Commercial License

- D'Fusion Studio Educational License (EDU)

○ I accept the terms of the License Agreement
● I do not accept the terms of the License Agreement

Figure 12: D'Fusion @Home Desktop License Agreement

It is mandatory to present the License Agreement to the final user. Third-party licenses must also be included.

5.2.1.2 Installer and Package redistribuable Microsoft Visual C++ 2010

The D'Fusion @Home Player needs the Package redistribuable Microsoft Visual C++ 2010 (x86), called "vcredist". The "vcredist_x86.exe" installer required version has been copied in the D'Fusion CDRom, in the subfolder called "redist".

Source to download the required vcredist is on the official Microsoft download center:

http://www.microsoft.com/downloads/fr-fr/confirmation.aspx?FamilyID=a7b7a05e-6de6-4d3a-a423-37bf0912db84

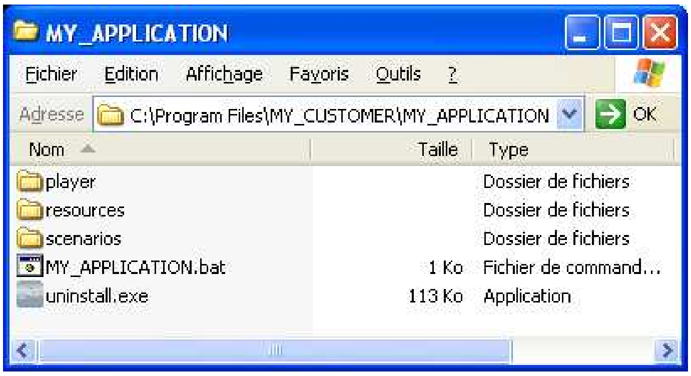
Please ensure your D'Fusion @Home desktop application installer detects if the installation of the Microsoft Visual vcredist 2010 is required, and install it if needed.

A way to do so could be to call the DFusionHomeDesktopPlayerSetup.exe installer, provided in the DFusion CDRom: this will install the DFusion Home Desktop Player and manage the vcredist installation.

5.2.1.3 Install directory and shortcut

The final D'Fusion @Home Desktop application can be launched in two ways. If the final scenario is placed near the executable, it can be directly executed. If the final scenario is placed in a folder different from the application folder, then the player must be launched in command line mode or by a shortcut and the scenario file path must be specified by using the –f argument.

Most often, the installation directory has the following structure:



Adresse C:\Program Files\MY_CUSTOMER\MY_APPLICATION

Nom	Taille	Type
player		Dossier de fichiers
resources		Dossier de fichiers
scenarios		Dossier de fichiers
MY_APPLICATION.bat	1 Ko	Fichier de command…
uninstall.exe	113 Ko	Application

Figure 13: D'Fusion @Home Desktop Install Directory

5.2.2 Standalone packaging on MacOSX platform

5.2.2.1 Placing the final scenario in the Contents Application folder

With Mac OS X operating system, applications may include in the application bundle all the resources, libraries and frameworks that are needed. It is, consequently, recommended to put the exported scenario folder with the content final software key into the DFusion@HomePlayer bundle.

The scenario folder content has to be placed into the "Scenario" folder, into the application bundle. To do this, you open the DFusion@HomePlayer package content.

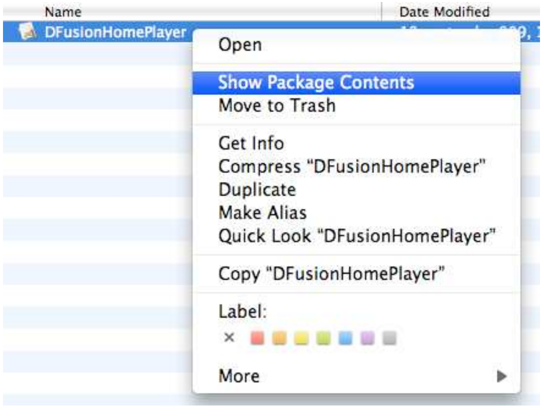


Figure 14: Show MacOSX package contents command

When the package content is opened, the exported scenario folder content with final software key must be placed in a "Scenario" folder, placed in the "Contents" folder.

"Scenario" folder name is mandatory. The DFusion@HomePlayer loads the first file with the "dpd" extension in "Scenario" folder as the scenario to play.

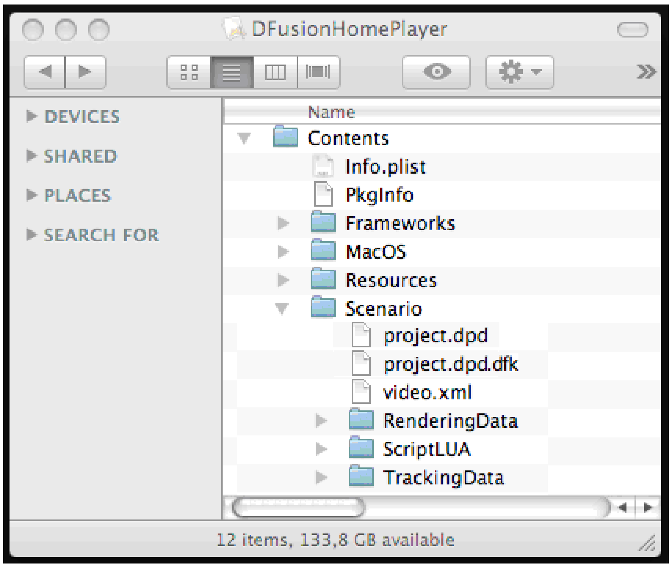


Figure 15: Copy Scenario in Contents folder

5.2.2.2 Changing the icon

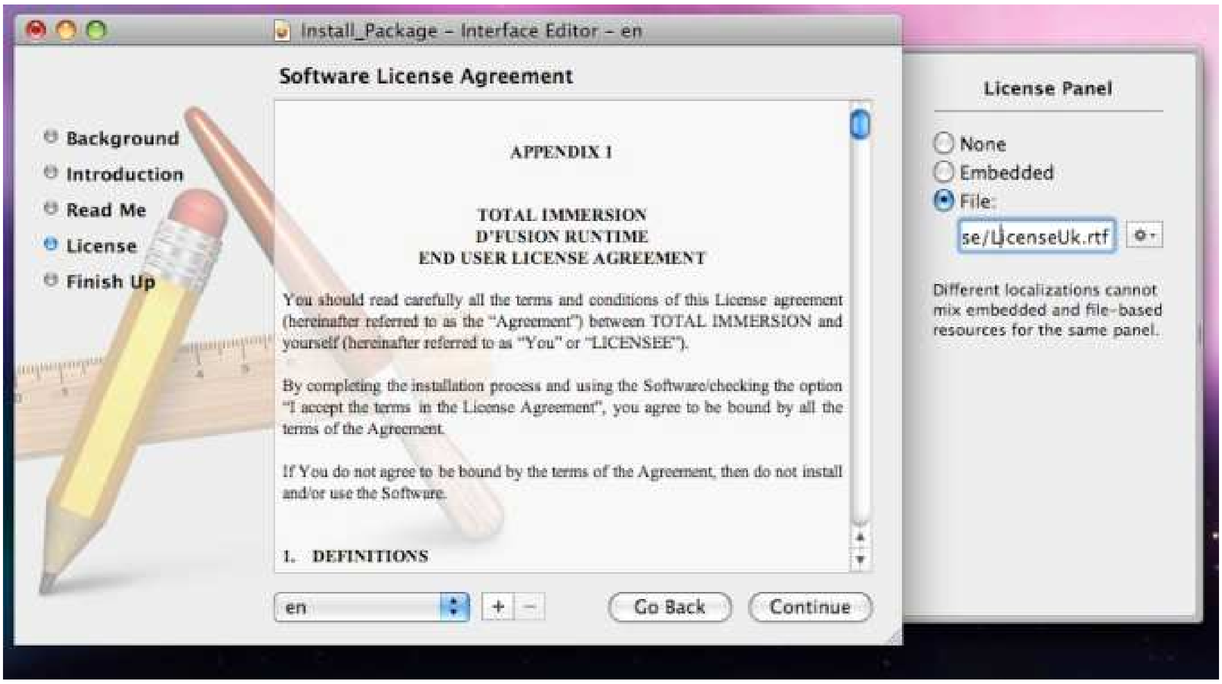
The application icon can be customized. You can change it by replacing the DFusion@Home icon in the Contents/Resources folder with your own icon. If you change the icon name, you must also update the Info.plist with the correct icon name.

5.2.2.3 Package the application

Apple provides an application named PackageMaker with its developer utilities. There are other ways to create installation packages, but we recommend using PackageMaker.

It is mandatory to present the License Agreement to the final user.

PackageMaker allow the easy creation of an installer for Mac OS X and to add a "read me" and "License" step.



APPENDIX 1

TOTAL IMMERSION
D'FUSION RUNTIME
END USER LICENSE AGREEMENT

You should read carefully all the terms and conditions of this License agreement (hereinafter referred to as the "Agreement") between TOTAL IMMERSION and yourself (hereinafter referred to as "You" or "LICENSEE").

By completing the installation process and using the Software/checking the option "I accept the terms in the License Agreement", you agree to be bound by all the terms of the Agreement.

If You do not agree to be bound by the terms of the Agreement, then do not install and/or use the Software.

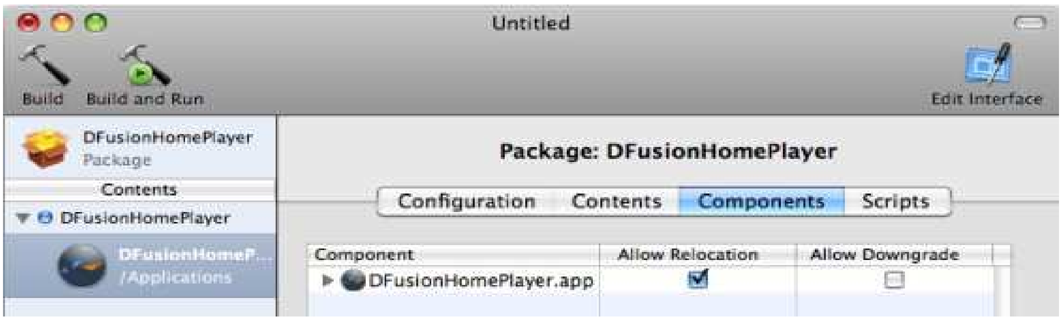
1. DEFINITIONS

○ None
○ Embedded
● File: se/LicenseUk.rtf

Different localizations cannot mix embedded and file-based resources for the same panel.

Figure 16: License agreement on Mac OS X

The relocation option must be unchecked; it will be more convenient and will secure reinstallation if the application is already installed.



Package: DFusionHomePlayer

Component	Allow Relocation	Allow Downgrade
DFusionHomePlayer.app	☑	□

Figure 17: Mac OS X installer - relocation

This document does not explain how to use PackageMaker, please refer to the PackageMaker User Guide.

When the installation package is created, we recommend using Disk Utility to create a disk image (.dmg file) from the installation package. Disk image ensure that binary integrity is preserved and is automatically mounted after being downloaded. Therefore, it simplifies the installation process for the final user.

6 ADVANCED

6.1 Splash Screen customization

In D'Fusion @Home, the splash screen is easily and highly customizable. It is composed of an .xml description file and some resources (image files). You will be able to play image sequences, define you own text areas, choose your background image, etc.

To launch your own splash screen (in testing versions), you have to use the -ssc command argument.

Example:

Your splash (ssc file + images) is contained in the "MyOwnSplash" folder which is at the same level as the scenario folder ("MyScenario") and at the same level as the D'Fusion @Home testing application folder ("Dfusion@HomePlayerTest"). To launch your scenario with your own splash screen in this application, you have to enter the command:

```
> DFusionDesktopPlayer.exe -ssc "..\MyOwnSplash\splash.ssc" -f "..\MyScenario\project.dpd"
```

or

```
> "@Home\DFusionDesktopPlayer.exe" -ssc "MyOwnSplash\splash.ssc" -f "MyScenario\project.dpd"
```

It depends of your current directory.

6.1.1 SSC file

The SSC file is an xml file that aims at describing the splash screen according to the downloading/loading scenario evolution. In some sections of the xml you will recognize this logic.

The main section of the file is "splash" with no attribute. It is just a starting point for the splash loader. In this section you have to declare at least three sub sections:

- window: it describes the splash screen window style
- progressbar: it describes the progress bar look and behavior through the different steps
- step: with its sub-sections it describes the splash screen behavior (images sequences, background image, text areas, etc.)

6.1.2 Window section

Attribute	Description
width	the width in pixel of the splash window
height	its height in pixel
transparency	can be "true" or "false". It controls the WS_EX_TRANSPARENT style of the window. If this value is set to "true", the flag is active and the backgroundcolor parameter is ignored
show	can be "true" or "false". Controls the visibility of the splash screen
backgroundcolor	a string containing RGB values of the background color of the window. Note that if a background image covers the entire splash window, the background color never appear

Example:

```
    <window      width=640
                 height=480
                 transparency=“false”
                 show=“true”
                 backgroundcolor=“105,140,189”/>
```

6.1.3 Step

In a step section, the user will describe the splash screen behavior during the scenario loading or in the scenario downloading (in the web case) with text and image sequences.

With the standalone player, you can only define one step section (sections declared afterwards will be ignored). With the web version, you can have two step sections: one corresponding to the downloading step, the other one corresponding to the scenario loading. This is not mandatory.

In each step you can define your own customized dynamic content like the progress bar, image and text sequences. The following sections describe how to configure it.

6.1.4 Progress bar

Attribute	Description
x	indicates, in pixels, the x (horizontal) coordinate of the upper-left corner of the progress bar relative to the upper-left corner of the splash window
y	same as x, for the vertical coordinate
width	the width in pixels of the progress bar
height	its height in pixels
backgroundcolor	string containing RGB values of the background color of the progress bar
showbackground	can be “true” or “false”. If the “false” value is set, the background rectangle (not the status bar) will be transparent
show	can be “true” or “false”
color	a string containing RGB values of the progress bar
evol_param	can be “time” or “progress” It describes how the progress bar will evolve. The “progress” value is available on both steps and the percentage progression is the percentage downloaded or an arbitrary loading percentage status.
loop	Only active if evol_param=“time” indicates if the progress bar loops.
duration	Only active if evol_param=“time” indicates the sequence duration in seconds

Example:

```
<progressbar x=228 y=193 width=325 height=10 background=“255,255,255” showbackground=“true”
show=“true” color=“7,224,250”/>
```

6.1.5 Image sequence

Attribute	Description
ID	a string that corresponds to the name you want give to the sequence
evol_param	can be "time" or "progress" It describes how the sequence will evolve. The "progress" value is only available in a "downloading" step and the percentage progression is the percentage downloaded. This attribute has an impact on the evolutionstep attribute (in an image section) interpretation
loop	only active if evol_param="time" indicates if the sequence loops
Duration	only active if evol_param="time" indicates the sequence duration in seconds

Example:

```
<image_sequence          ID="foreground"
                         evol_param="time"
                         loop="true"
                         duration=2>
...
</image_sequence>
```

These attributes describe some global parameters of the sequence. The different image sub-sections help us to describe precise behavior of this sequence. An image sub-section describes one image, its size, its location, etc.

Attribute	Description
location	indicates the image path, relative to the ssc file
x	indicates, in pixels, the x (horizontal) coordinate of the upper-left corner of the displayed image relative to the upper-left corner of the splash window
y	same of x, for the vertical coordinate
scalex	note that we cannot control the width or height coordinates, because they are automatically calculated. However, you can adjust the rendered image size with scale (x or y parameters)
scaley	same of x, for the vertical coordinate
show	can be "true" or "false"
evolutionstep	a number that indicates when the splash screen has to pass to this <image>. If, in image_sequence section, the evol_param attibute is set to "progress", the evolutionstep attribute must be contained between 0 and 1 and corresponds to a percentage. If evol_param="time", then 0< evolutionstep value < image_sequence's duration attribute value and corresponds to a time in seconds
color	a string containing RGB values of the text

Example: A complete image_sequence description can look like this:

```
<image_sequence   ID="foreground"
                  evol_param="time"
                  loop="true"
                  duration=0.5>
          <image        location="fireball001.jpg" x=100 y=0
                        scalex=0.4 scaley=1 show="true"
                        evolutionstep=0.0 />
          <image        location="fireball002.jpg " evolutionstep=0.1 />
          <image        location="fireball003.jpg " evolutionstep=0.2 />
          <image        location="fireball004.jpg " evolutionstep=0.3 />
          <image        location="fireball005.jpg " evolutionstep=0.4 />
</image_sequence>
```

6.1.6 Text sequence

Creating text sequences is structured the same was as creating image sequences. So, for the sequence attributes, there is no difference. In text_sequence, you declare text sub-sections. Obviously, specific text attributes have to be set in text sub sections.

The attributes of a text sub-section are:

Attribute	Description
value	the text to be displayed
Font	Its font in a string ("Comic Sans MS" for xample).
size	the font size
Bold	can be "true" or "false"
italic	can be "true" or "false"
underlined	can be "true" or "false"
left, top, right and bottom	define, in pixels, a text area in the window. The text will be displayed at the center of this text area
color	a string containing RGB values of the text

Example:

```
<text_sequence ID="splash text">
          <text   value="text sample.."  font="Arial" size=16 bold="true" italic="false"
                  underlined=false" left=228 top=150 right=640 bottom=480 color="7,224,250"/>
</text_sequence>
```

6.1.7 Full sample

```
<?xml version=“1.0” standalone=“yes” ?>
<splash>
        <window         width=800 height=600 transparency=“false”
                        show=“true” backgroundcolor=“255,140,189”/>
        <progressbar    x=228 y=193 width=325 height=10 backgroundcolor=“255,255,255”
                        showbackground=“true” show=“true” color=“7,224,250”/>
        <step>
                <image_sequence ID=“background” evol_param=“time” loop=“true” duration=2>
                        <image          location=“splash.bmp” x=0 y=0 width=640 height=480
                                        scalex=1 scaley=1 show=“true”  evolutionstep=0.0 />
                        <image          evolutionstep=1.0 />
                </image_sequence>
                <image_sequence ID=“foreground” evol_param=“time” loop=“true” duration=1,5>
                        <image          location=“fireball001.jpg” x=100 y=0 width=800 height=600
                                        scalex=0.4 scaley=1 show=“true” evolutionstep=0.0 />
                        <image          location=“fireball005.jpg” evolutionstep=0.1 />
                        <image          location=“fireball010.jpg” evolutionstep=0.2 />
                        <image          location=“fireball015.jpg” evolutionstep=0.3 />
                        <image          location=“fireball020.jpg” evolutionstep=0.4 />
                        <image          location=“fireball025.jpg” evolutionstep=0.5/>
                </image_sequence>
                <text_sequence ID=“splash text”>
                        <text   value=“BITTE WARTEN SIE WAHREND TOPPS LIVE LADT… "
                                font=“Arial” size=16 bold=“true” italic=“false” underlined=“false”
                                left=228 top=150 right=640 bottom=480 color=“7,224,250”/>
                </text_sequence>
        </step>
</splash>
```

6.1.8 Some useful tips

- When declaring a sequence, you can omit attributes that have the same value in the previous entity. Like in the image_sequence example given in paragraph 2.3.1.
- You can force the redraw of an image by declaring it twice (or more) in the same sequence. It can solve draw issues in particular situations.

Example:

```
<image_sequence ID=“background” evol_param=“time” loop=“true” duration=2>
        <image          location=“splash.jpg” x=0 y=0 width=640 height=480
                        scalex=1 scaley=1 show=“true”  evolutionstep=0.0 />
        <image          location=“splash.jpg” evolutionstep=1/>
</image_sequence>
```

Here the background image is drawn every second.

6.1.9 Limitations

For the time being, it is advised to:

- not dynamically change location and scale for images in the same sequence
- have a background image and make the other sequences (text or image) be disconnected each other with no common area (except with the background image)
- not use evol_param=“progress” in steps which are not corresponding to the downloading scenario steps

- for now, it is better to use only bmp image files

6.2 Location of downloaded files and log files

- D'Fusion @Home Web Player: Scenario and log files

Under Windows Vista and Windows Seven, web browsers can run under a **protected mode** or a **non protected mode.** Internet Explorer can be executed under a protected mode or a non protected mode. All other web browser (Mozilla Firefox, Safari …) are always executed under a non protected mode.

Operating System	Web Browser	Kind of files	Path
Windows XP	Any	Scenario	`%APPDATA%\Total Immersion\Web`
	Any	Logs	`%TEMP%\Total Immersion\WebLog`
Windows Vista and Seven	Protected Mode (IE)	Scenario	`C:\Users\{PUT_USER_NAME_HERE}\AppData\Local\Microsoft\Windows\Temporary Internet Files\Virtualized\C:\Users\{PUT_USER_NAME_HERE}\AppData\Roaming\Total Immersion\Web`
	Protected Mode (IE)	Logs	`C:\Users\{PUT_USER_NAME_HERE}\AppData\Local\Temp\Low\Total Immersion\WebLog`
	Protected Mode (IE)	Download	`C:\Users\{PUT_USER_NAME_HERE}\AppData\Local\Microsoft\Windows\Temporary Internet Files\Virtualized\{PUT_DOWNLOAD_PATH_HERE_REPLACING_:_BY_\}`
	Non Protected Mode	Scenario	`C:\Users\{PUT_USER_NAME_HERE}\AppData\Roaming\Total Immersion\Web`
	Non Protected Mode	Logs	`C:\Users\{PUT_USER_NAME_HERE}\AppData\Local\Temp\Total Immersion\WebLog`
	Non Protected Mode	Download	`{DOWNLOAD_PATH}`
MacOS English	Any	Scenario	`~/Library/Application Support/Total Immersion/Web`
	Any	Logs	`~/Library/Caches/TemporaryItems/Total Immersion/`

- D'Fusion @Home Desktop Player: log files

Operating System	Path
Windows XP / Vista / Seven	%TEMP%\Total Immersion\Log
MacOS English	~/Library/Caches/TemporaryItems/Total Immersion/Log

6.3 How to uninstall the D'Fusion @Home Web Plug-in on MacOS

Erase these files manually ("Move to Trash" on English MacOS):

```
/Library/Receipts/DFusionWeb.Installer.pkg
/Library/Receipts/DFusionHomeWebPlugin.Installer.pkg
/Library/Internet Plug-Ins/DFusionWeb.plugin
/Library/Internet Plug-Ins/tiGeckoPluginDFusion.xpt
/Users/username/Library/Application Support/Total Immersion
```

When you try to install an older version (for example installing 2.20.1000 after having installed and uninstalled 2.20.2000) it is possible the plug-in installation fails. The setup will say something like "a newer version of this package is already present on the system".

When this problem occurs, the only solution (at this time) is to use the **pkgutil** (man page) command line tools.

- Use ***pkgutil –pkgs*** to list all package on the system.
- TI packages identifiers begin with com.t-immersion.
- Then use ***pkgutil –forget com.t-immersion.XXX*** to remove your package from the MacOSX package database. Note that to run this command you have to be ***root***. It means you have to execute this command using **sudo** (or at least **su**).

6.4 How to detect if D'Fusion @Home Web Plug-ins are installed

To detect if the D'Fusion @Home Web Plug-in needs to be installed or updated, you simply have to use the function **tiTestDFusionPlugin()** (from the file totalimmersion_dfusion_web.js). This function returns a number to determine the state of the plug-in:

- 0: the plug-in is up to date.
- 1: the plug-in needs to be installed. It means no version of the plug-in has been found on the end-user system).
- 2: the plug-in need to be updated. It means the local version of the plug-in (the one currently installed on the end-user system) is older than the one required by the web page.

Remarks:

- Under Apple Mac OS X with Mozilla Firefox the plug-in version is not tested (at this time). It means only 0 or 1 can be returned.
- Under Windows with Microsoft Internet Explorer the ActiveX version is not tested (at this time). It means only 0 or 1 can be returned. But not that Microsoft Internet Explorer has its own mechanism to install ActiveX so the end user will be automatically notified when the ActiveX need to be (re)installed.

6.5 How to generate Test Software Keys with tiProtectorAR2

The tiProtectorAR2 tool is available with D'Fusion Studio Suite Pro only. You will find it in the *DFusion Studio Pro installation folder* > *Deployment* > *Tools* folder.

You will use tiProtectorAR2 to generate a Test Software Key (and make an optional complete encrypted package) of a modifiable/dynamical part of your project.

One typical use case is when you want to export and generate a Test Software Key for a folder that is to be downloaded from the internet, and dynamically loaded and used with your master (main) AR scenario at execution time depending on the final user actions. Every data used into your project has to be signed (i.e. associated to a Test Software Key). For downloaded data to be used correctly it has to be signed into the main scenario Software Key (generated when exporting your scenario) and it has to be accompanied by its own signature (generated with tiProtectorAR2).

This section describes how to create a Test Software Key(s) for specific data using the tiProtectorAR2 (command line) tool.

6.5.1 Using tiProtectorAR2

To use tiProtectorAR2 for D'Fusion @Home, you must execute the following command line:

```
tiProtectorAR2.exe parameters.xml [outfolder] –t @Home [-p Platform] [-e Entrypoint ] -slave
```

The following table reviews the arguments used above:

Arguments of the tiProtectorAR2.exe		
Mandatory	**Argument**	**Comment**
YES	`parameters.xml`	This is the name of an XML file containing the description of the data to download. A sample type is provided below.
YES	`outfolder`	Folder where all the data to download is to be exported. The software key generated by tiProtectorAR2 will also be placed here.
YES	`-t @Home`	Option specifying that you're creating a Software Key for an @Home target.

Other variables that can be used with the tiProtectorAR2 tool, as well as a sample type for "parameters.xml", are described in the following section. @Home-specific use cases can be found in section "6.5.4 Use case examples".

6.5.2 tiProtectorAR2 arguments

- **parameters.xml**: this is the complete path (absolute or relative) to the xml file that references all the data to export, and for which a Test Key is to be generated. You must create this file from scratch with a key-exporting compatible configuration, as shown in the example below:

```
<dfusion version="3.20.xxxxx">
   <configurations>
      <configuration name="@Home|All Platforms" encryption="none" />
      <configuration name="mobile|All Platforms" encryption="none" />
      <configuration name="mobile|android" encryption="none" />
      <configuration name="mobile|iPhoneOS" encryption="none" />
   </configurations>
   <settings>
      <application applicationid="mobile|android:appId; mobile|iPhoneOS:appId " title="" icon="" />
   </settings>
   <resources>
      <files>
         <file relativePath=".\file1.mesh" />
         <file relativePath=".\file2.scene" />
          .....
      </files>
   </resources>
</dfusion>
```

Each tag specified in the parameters.xml file is described below:

<configurations> <configuration>	
Attribute	**Description**
name	String description of a *"Target|Platform"* configuration
encryption	String that defines the encryption process to apply when the configuration is exported

<settings> <application>	
Attribute	**Description**
applicationid	String that describes the application IDs for each configuration defined in <configurations> *Target|Platform:applicationID;Target2|Platform2:applicationID2*

<resources> <files><file>	
Attribute	**Description**
relativePath	File path to the data to generate a test key for, relative to the file "parameters.xml"

- **outfolder**: Optional (but recommended). If specified, the files that are listed in your "xml" file are exported into this folder, and encrypted according to the encryption specified for the selected *Target* and *Platform* (see next optional arguments). If not specified, the Test Software Key is generated in the folder where your "xml" file is contained, and there is no encryption.

- **-e Entrypoint:** Optional. If specified, for example "-e car.scene", the generated Software Key will be named "car.scene.dfk", and the FIRST media to load from the exported folder must be the "car.scene" media.

- **-t Target**: Optional. If not specified, the first <configuration> specified in the <configurations> list is used. If specified, it corresponds to the *Target* for which the export has to be done. This Target needs to be listed in the <configurations> tag. Possible values are:

@Home	mobile	Pro

- **-p Platform**: Optional. It corresponds to the *Platform* for which the export has to be done.

	-p Platform specified	*-p Platform* NOT specified
-t Target specified	The first configuration in parameters.xml that matches the specified *Target* and *Platform* is used	The first configuration in parameters.xml that matches the specified *Target* is used
-t Target NOT specified	The first configuration in parameters.xml is used	The first configuration in parameters.xml is used

Possible values for the Platform argument are listed below:

Target	@Home	mobile	Pro
Platform	All Platforms windows macOSX	All Platforms android iPhoneOS	All Platforms windows macOSX

- **-slave**: Mandatory for data to download dynamically. If not specified, the Software Key is marked "master". If "-slave" is specified, the Software Key is marked "slave". "Slave" indicates that the set of files for which the test key is to be generated are part of a bigger project, typically the files (media) to download dynamically from a webserver.

- The "-slave" option must be set for data to download dynamically.
- A "master" scenario can ONLY use external data signed as "slave".
- A "master" scenario can NOT use external data signed as "master".
- A D'Fusion runtime can not load a project signed as "slave".

6.5.3 Generated keys naming conventions

The following points describe the naming conventions used for keys generated with the tiProtectorAR2 tool:

✓ If the input xml file is a file named "myinput.xml", the name of the output Test Software Key is "myinput*.xml.dfk"*

✓ If the input xml file is an xml file named for example "myinput.xml", and the *outfolder* option is set to "*MyFolderToDownload*",

  o If no "entry point" is defined in the "–e" argument, then the key name is "*MyFolderToDOwnload.dfk*",

  o If the –e "*Entrypoint*" argument is specified, then the key name is "*Entrypoint.dfk*"

Other tiProtectorAR2 arguments combinations are not recommended.

6.5.4 Use case examples

See below a sample parameters.xml file that corresponds to the following “MyFolderToDDownload” folder:

```
+ MyFolderToDownload
  - parameters.xml        ------>   NOT to be exported
  - file1.scene           ------>   to be exported
  - file2.mesh            ------>   to be exported
  - file3.jpg             ------>   to be exported
  + subFolder1            ------>   Empty folder cannot be exported
    - file4.mesh          ------>   to be exported
    - file5.jpg           ------>   to be exported
    + subFolder1.2        ------>   Empty folder cannot be exported
      - file6.mesh        ------>   to be exported
      - file7.jpg         ------>   to be exported
  + subFolder2            ------>   Empty folder cannot be exported
    - …                   ------>   NOT to be exported
```

According to the instructions above, your “parameters.xml” should contain the following code:

```
    <dfusion version=“3.20.xxxxx”>
      <configurations>
        <configuration name=“@Home|All Platforms” encryption=“encrypted” />
        <configuration name=“@Home|windows” encryption=“encrypted” />
        <configuration name=“@Home|macOSX” encryption=“encrypted” />
        <configuration name=“mobile|All Platforms” encryption=“encrypted” />
        <configuration name=“mobile|android” encryption=“encrypted” />
        <configuration name=“mobile|iPhoneOS” encryption=“encrypted” />
      </configurations>
      <settings>
        <application applicationid=“mobile|android:com.appid.testapp; mobile|iPhoneOS:com.appid.testapp ”
title=“” icon=“” />
      </settings>
      <resources>
        <files>
          <file relativePath=“.\file1.scene” />
          <file relativePath=“.\file2.mesh” />
          <file relativePath=“.\file3.jpg” />
          <file relativePath=“.\subFolder1\file4.mesh” />
          <file relativePath=“.\subFolder1\file5.jpg” />
          <file relativePath=“.\subFolder1\subFolder1.2\file6.mesh” />
          <file relativePath=“.\subFolder1\subFolder1.2\file7.jpg” />
        </files>
      </resources>
    </dfusion>
```

Be carreful: The player rendering engine, Ogre3D, can use downloaded resources only from newly created folders. CONSEQUENTLY, A SCENARIO MUST DOWNLOAD ADDITIONAL DATA INTO A NEWLY CREATED FOLDER ON THE USER COMPUTER.

6.5.4.1 Typical use 1 (RECOMMENDED)

When you use the tiProtectorAR2 tool to export and sign the folder to download, you can use the following command line:

```
tiProtectorAR2.exe MyFolderToDownload\parameters.xml FinalOutName -t @Home –p windows -slave
```

The result of the previous command line execution is:

```
+ FinalOutName
- FinalOutName.dfk
- file1.scene
- file2.mesh
- file3.jpg
+ subFolder1
  - file4.mesh
  - file5.jpg
  + subFolder1.2
    - file6.mesh
    - file7.jpg
```

- ✓ **NO CHANGE ALLOWED on the signed files.**
- ✓ **Key name = Parent folder name + ".dfk"**
- ✓ **First media to load = any media**

In our example, the parent folder name of the key must always be "**FinalOutName**" we will have to download

- from *http://www.mysite.com/…/**FinalOutName***
- to the user computer into *<User App Directory>/…/**FinalOutName***

6.5.4.2 Typical use 2

When you use the tiProtectorAR2 tool to export and sign the folder to download, you can use the following command line:

```
tiProtectorAR2.exe MyFolderToDownload\parameters.xml FinalOutName -t @Home –p windows –e file1.scene -slave
```

The result of the previous command line execution is:

```
+ FinalOutName
  - file1.scene.dfk
  - file1.scene
  - file2.mesh
  - file3.jpg
  + subFolder1
    - file4.mesh
    - file5.jpg
    + subFolder1.2
      - file6.mesh
      - file7.jpg
```

- ✓ **NO CHANGE ALLOWED on the signed files.**
- ✓ **Key name = Media name + ".dfk"**
- ✓ **First media to load = THE Media**

No constraint on folder names, on the webserver or on the user computer once downloaded.

In our example, the first media that the scenario must load from the downloaded folder is "file1.scene".

Be careful: The rendering engine part of the player, Ogre3D, can use downloaded resources only from newly created folders. CONSEQUENTLY, A SCENARIO MUST DOWNLOAD ADDITIONAL DATA INTO A NEWLY CREATED FOLDER ON THE USER COMPUTER.

6.6 License coherency using multiple Software Keys

A Software Key can be in three different modes, depending on the tool used to generate it:

- "Test"
- "Test Free"
- "Final"

D'Fusion Studio generates a "Test Free" Software Key when exporting a project.

D'Fusion Studio Pro and the tiProtectorAR tool generate "Test" Software Keys.

A "Test" Software Key can be converted to "Final". "Final" Software Keys will allow you to deploy your application for commercial use.

As explained in the previous chapter of the current document, a *Master* key is a key that can be loaded firstly. A *Slave* key corresponds to data that can be added dynamically to the master project, and loaded in second time (after a download for example).

License coherency between Software Key modes			
Master "Test" with …	**Slave "Test"**	~~Slave "Test Free"~~	~~Slave "Final~~
Master "Test Free" with …	~~Slave "Test"~~	**Slave "Test Free"**	~~Slave "Final"~~
Master "Final" with …	~~Slave "Test"~~	~~Slave "Test Free"~~	**Slave "Final"**

7 USE CASE: WEB SOLUTION

7.1 How to send commands from JavaScript to the player

In the *.js files provided with the web application, you can find an "ExecuteCommand(command)" javascript function. This function binds directly the "command" argument to the web plug-in. This single argument have to be a string formatted like this:

```
        function_name ' arg0' 'arg1' 'arg2' ....
```

Example:

```
        function MakeKeystrokeActions(){
          for (var i = 0; i < arguments.length; i++) {
                ExecuteCommand("PLAYWITHKEY '"+arguments[i]+"'");
          }
        }
```

This function can be called like this: MakeKeystrokeActions("y", "a", "m");

For the function_name argument part there is two options:

- It refers to a command known by the plug-in itself. So, this command is processed in the plug-in. For now, the available commands embed in the plug-in are: playwithkey, camselected and gofullscreen (not case sensitive).
- The command is unknown from the plug-in, so it's dispatched to the player.

In Lite players, commands gesture is defined within the project code specific part. In each project, you have to edit the "ExecuteCommand" function to handle your custom commands.

In the generic AR2 @Home player, your commands are bind to LUA such a manner that you just have to write some LUA script to handle.

In your Script you have to declare a webInterface object and get commands from web plug-in:

```
        local webInterface = getWebInterface()
        local isCommand = 0
        local command = 0
        isCommand, command = webInterface:pullWebCommand()

        if isCommand then
                if command["CommandName"]=="SCALE" then
                        geom:getScale(scale)
                        scale:setX(command["arg0"])
                        if command["arg1"] then
                                scale:setY(command["arg1"])
                        else
                                scale:setY(command["arg0"])
                        end
                        if command["arg2"] then
                                scale:setZ(command["arg2"])
                        else
                                scale:setZ(command["arg0"])
                        end
                end
        end
```

As you have noticed, the webInterface:pullWebCommand() function returns two values: a boolean (that indicates if an available web command have been sent from the plug-in) and a table (that contains the command name and all its arguments).

7.2 How to execute JavaScript functions from a Lua script

Here again, you need the webInterface LUA object:

```
local webInterface = getWebInterface()
```

This time, this is the execJScript function that applies.

Example:

```
local text = string.format(        "scaleX=%g, scaleY=%g,
                                   scaleZ=%g", scale:getX(),
                                   scale:getY(), scale:getZ())
webInterface:execJScript( "WriteText", text )
```

All of the parameters have to be strings. The first param is the name of the javascript functin to call. All others are its arguments.

8 TROUBLESHOOTING

8.1 Plug-in backward compatibility

Since D'Fusion 2.1.8814 an important bug in the plug-in is fixed.

Before the 2.1.8814 release, if you plaid an application which required 2.1.X and you never installed any TI Web Player, the plug-in 2.1.X was downloaded and installed. Then, if you visited an application which required DFusionAR2 Web 2.1.Y, (Y < X), since release is older (Y < X), the plug-in didn't update the player as you already have the 2.1.X. But there was a bug in the plug-in and it was never notified that the update was finished so it was always waiting for it.

With the new release everything is fixed. But to really solve the problem on applications already released, you have to update these applications' Web server: only the plug-in and the JavaScript file have to be updated.

You will find files to update in the "online\plugin" folder of the D'Fusion Web package:

```
DFusionHomeWebPlugIn.Installer.dmg
DFusionHomeWebPlugIn.Installer.exe
totalimmersion_dfusion_web_plugin.js
totalimmersion_dfusion_web_pluginMac.js
```

8.2 Downloading problem

The binary mode for download has to be ensured by the web server, this has to be configured for all files, even if their extension corresponds to a textual file.

Files have to be accessible for download.

To prevent any future problems, please ensure that this is supported and actually functional on your hosting web server.

Note that these points are really important and if not ensured the D'Fusion Web application will not function at all, because either none files could be found on the server by the D'Fusion Web application, or, if found, corrupted files could be downloaded.

8.2.1 Issues to be prevented with Apache

- Apache on Windows: a priori, there is no problem with this configuration
- Apache on Unix/Linux: ensure that the whole copied directory and its subdirectories have the execution ("X") right set, so that the Apache server can step into. If not, the D'Fusion Web download will fail.

8.2.2 Issues to be prevented with IIS

You could (and probably will) encounter problems when the hosting server is using Microsoft IIS:

- All files that are hidden by default by Windows (those beginning with "."), especially all occurrences of the ".update.xml", need to be marked as non-hidden: just uncheck the hidden attribute. If not, the ".update.xml" file will not be downloadable because IIS is shielding all hidden files.
- You'll need to ensure that all application files you're placing onto the server (Plugin/Player/Scenario) are correctly registered for a binary download. That is to say, you'll need to ensure that the following MIME type is associated with their extension (or with all extensions via the ".*" wildcard):

application/octet-stream

For IIS, ensure that the MIME type for the whole project is correctly configured. To do that, click "Properties" on your D'Fusion Web project directory within your IIS WebSite configuration:

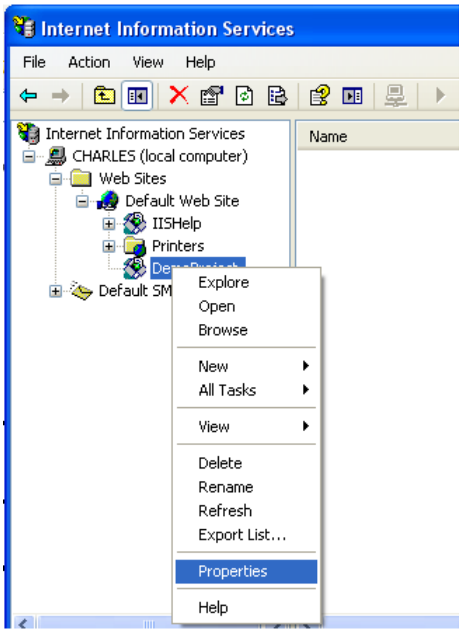
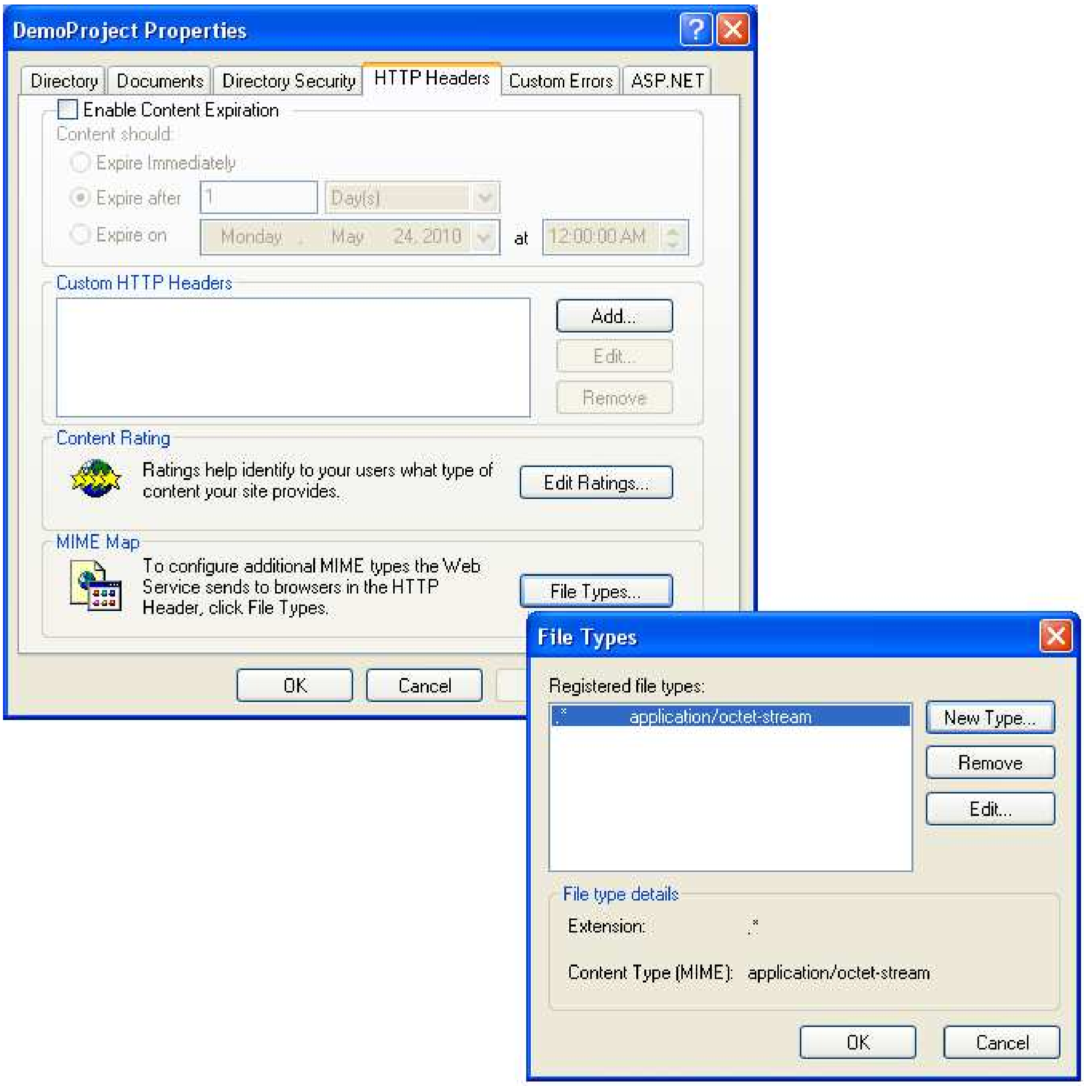


Figure 18: IIS Web site configuration

Then, within the "HTTP headers", choose the "Types MIME" and configure them as shown below:



DemoProject Properties

Directory | Documents | Directory Security | HTTP Headers | Custom Errors | ASP.NET

□ Enable Content Expiration

Content should:

○ Expire Immediately

● Expire after 1 Day(s)

○ Expire on Monday , May 24, 2010 at 12:00:00 AM

Custom HTTP Headers

Add... Edit... Remove

Content Rating

Ratings help identify to your users what type of content your site provides.

Edit Ratings...

MIME Map

To configure additional MIME types the Web Service sends to browsers in the HTTP Header, click File Types.

File Types...

OK Cancel

File Types

Registered file types:

.* application/octet-stream

New Type... Remove Edit...

File type details

Extension: .*

Content Type (MIME): application/octet-stream

OK Cancel

Figure 19: IIS MIME types configuration

If you do not prefer to configure whole your site via ".*" wildcard, you'll need to configure case-by-case for all file types present in the player's and scenario's directory.

Note also, that in the Mac declination of the D'Fusion @Home Web product, there are files without extension. Those need to be also correctly configured for a binary download under IIS. This is done by using "." (one dot) as extension.

The following table presents file extensions that are not defined by default in the IIS. Thus, to ensure correct download functionality, they need to be added manually with the corresponding MIME-type:

Precise IIS MIME type configuration	
Extension mask	**MIME type**
.cg	application/octet-stream
.dpd	application/octet-stream
.fontdef	application/octet-stream
.lua	application/octet-stream
.material	application/octet-stream
.mesh	application/octet-stream
.glsl	application/octet-stream
.mp4	video/mp4
.nib	application/octet-stream
.ogg	application/octet-stream
.overlay	application/octet-stream
.particle	application/octet-stream
.plist	application/octet-stream
.program	application/octet-stream
.scene	application/octet-stream
.skeleton	application/octet-stream
.ssc	application/octet-stream
.strings	application/octet-stream
.tga	application/octet-stream
.xib	application/octet-stream
.	application/octet-stream

8.2.3 Microsoft Visual Redist cannot be downloaded

Sometimes, in rare use case, users failed to install the application because a firewall prevented Microsoft Visual redist from being downloaded. Consequently the application cannot be done correctly.

A solution is to provide a separate link in your web page that will allow downloading the installer version that contains these files and correctly installing the application. The installer name is "DFusionHomeWebPlugIn.InstallerFull.exe" and is located in the DFusionAR2Web\online\plugin folder of the cdrom. Be aware that this version weights 3,4 Mo, and that downloading will take more time.

8.3 @Home Web player loading problem

8.3.1 Div section must not be hidden

- Description: the Firefox or IE plug-ins is correctly downloaded and installed, the D'Fusion @Home Web player is correctly downloaded but cannot be launched.
- Solution: In the HTML page, the "div" section used for the web player must not be "hidden", otherwise the player is unloaded.

8.4 Flickering problem with Internet Explorer

- Description

  Within an iFrame, you're facing some flickering problem with the background of the iFrame and the video, while D'Fusion is running under IE (problem seems to appear most of the time only with IE6). You also see that the flickering disappear as soon as the scrolling bar is used or the window is resized.
- Solution

  Write a JavaScript function to be called when the player has launched the scenario. This function will be executed only once (test the return value to trigger it again if failed during the next frame) as follow:

```
function ActivateScenario()
{
        top.window.focus();
}
```

8.5 Tracking Problems

The 3D doesn't appear on the interactive pages that I am presenting to the camera.

The success of the experience depends both on the lighting conditions in which you carry it out and the manner in which you present the pages to the camera.

Make sure the pages are well lit, making certain that there are no shadows or reflections.

Present the page in front of the camera, in a way that the page takes up half of the video

Verify that your camera is not configured in mode mirror or symmetry hexagonal.

8.5.1 I have got inverse 3D

Sometimes there is scene inversion (tilted downwards when the panel is upwards). To rectify this, place the panel vertically opposite the camera and then incline it once more.

This problem is also connected with lighting. Increase the lighting, avoid back light and this problem will be reduced.

8.5.2 How to improve the lighting

Avoid back lighting and insufficient lighting, it creates problems because the software has difficulty locating the tracked object. Lighting should be turned up to improve this. If you cannot improve the lighting in the room, add lighting to the top of the terminal.

8.6 Rendering Problems

8.6.1 On some computers, materials are white instead of being textured

It has been noticed that on one computer, all the 3D objects were blank. This computer was powerful enough for the project requirements. Exactly the same project did not produce this effect on another computer.

This issue comes from the light parameters, precisely from the attenuation. In the project file (*.dpd), there was:

```
attenuation="{100000, 0, 0, 0}" //"{range, constant, linear, quadratic}"
```

In the Wiki section of Ogre3D, the light attenuation formula is described as following:

Attenuation = Constant + Linear * Distance + Quadratic * Distance^2

Attenuation = 0 + 0*distance + 0*distance^2 = 0

And

Luminosity = 1 / Attenuation

There is here a division by 0!

So it leads, on some graphic cards, to a fully blank material.

The solution to avoid this problem is to never put the "constant" parameter at 0.

8.6.2 On MacOSX, particles are not correctly rendered

It has been noticed that on some mac computer, the particles are not correctly rendered. The problem comes from the MacOSX Ogre platform.

Actually we have noticed 3 particle parameters causing troubles:

- The emitter type *must* be set to "Point"
- The affector "Scaler" *must* not be used with a negative scale.
- The field "point_rendering" must be set to "false" to have the same rendering on a windows and on a MacOSX platform.

A particle system file (*.particle) sample content:

```
particle_system sample
{
        ...
        point_rendering  false
        ...
        emitter Point
        {
        ...
        }
}
```

8.6.3 On some Mac, textures are not correctly displayed

It has been noticed that on some mac computer, the textures are not correctly displayed. The problem comes from the MacOSX Ogre platform.

Verify that the texture has a power of 2 resolution (128x128, 256x256, …), if not:

- The material referencing it must have mipmap activated (not 0)
- If it is an texture dynamically created with lua script, call setNumMipmaps() before serRessource().

8.7 Recording Problems

8.7.1 Video and Audio recording problems

- *Recorded Audio has Lag*

It has been noticed that on some computer, the Video/audio recording has some lags in the audio. The problem happens when using custom WMV profile (a lot of WMV profiles seem to have this problem).

The solution to avoid this problem is to use the method:

setVideoRecordConfig (string filePath, string type, int targetW, int targetH, float videoRate, int videoCPU, int videoBitRate, bool recordAudio)

which has a default WMV profile that always works.

- *Captured sound always recorded*.

The difference between a file sound and a captured sound (microphone for example) is that captured sound is always recorded when a video is recorded even if it is not played (played by speakers).

If you do not want to record the captured sound, then you must make it silent by using the method **setGain(0.0)** on the captured sound. The sound will still be captured (capturing microphone input), but won't be recorded.

8.8 Application crashed

8.8.1 Graphic card does not support more than 2 lights

Some old computers with a configuration inferior or equal to minimum hardware requirements have a graphic card that cannot support more than 2 virtual lights. The solution is to reduce the number of virtual lights and to keep only the main source lights.

8.9 Internet Explorer Protected Mode

Internet Explorer (7 and >) introduced Protected Mode, a feature which helps ensure that the browser and its add-ons run with a minimal set of permissions. To help ensure compatibility, Protected Mode employs a system of virtualization so that code running within Protected Mode continue to work even if its permissions are restricted.

However, if user switches between modes, D'Fusion Web plugin may encounter some issues.

To be able to use the plugin under **IE Protected Mode**, it must be first installed under that configuration. Another installation may be asked for using it under **Firefox** or **IE non Protected**. Doing that way, installation will be proper.

Otherwise if user first installs D'Fusion Web plugin under Firefox or **IE non Protected**, installation will not be functional for **IE Protected** mode and uninstallation is then requisite.

Note: Here's how to enable/disable protected mode:

- Open Internet Explorer.
- From the Internet Explorer command bar, choose Tools and then Internet Options.
- In the Internet Options window, click on the Security tab.
- Below the Security level for this zone area, and directly above the Custom level... and Default level buttons, uncheck the Enable Protected Mode checkbox.
- Note: Disabling Protected Mode requires a restart of Internet Explorer, as you may have seen next to the checkbox in this step.
- Click OK on the Internet Options window.
- If you're prompted with a Warning! dialog box, advising that The current security settings will put your computer at risk., click the OK button.
- Close Internet Explorer and then open it up again.